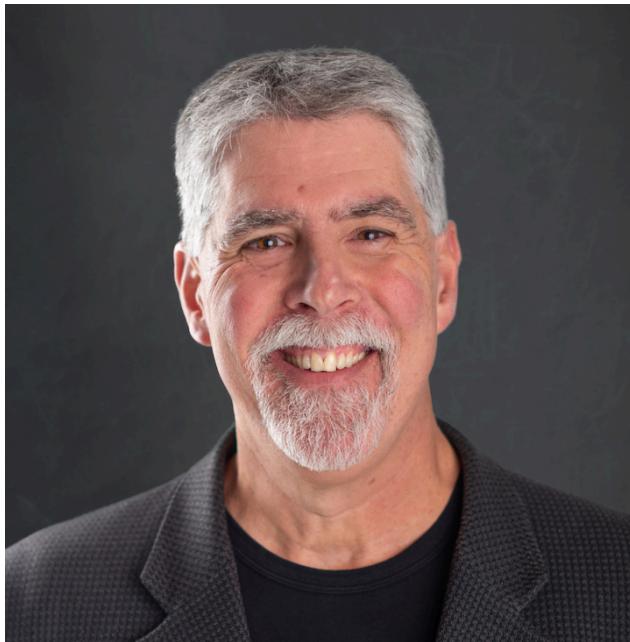


# Software Architecture by Example



## Mark Richards

Independent Consultant

Hands-on Software Architect, Published Author

Founder, [DeveloperToArchitect.com](http://DeveloperToArchitect.com)

<http://www.wmrichards.com>

@markrichardssa



## Neal Ford

ThoughtWorks

Director / Software Architect / Meme Wrangler

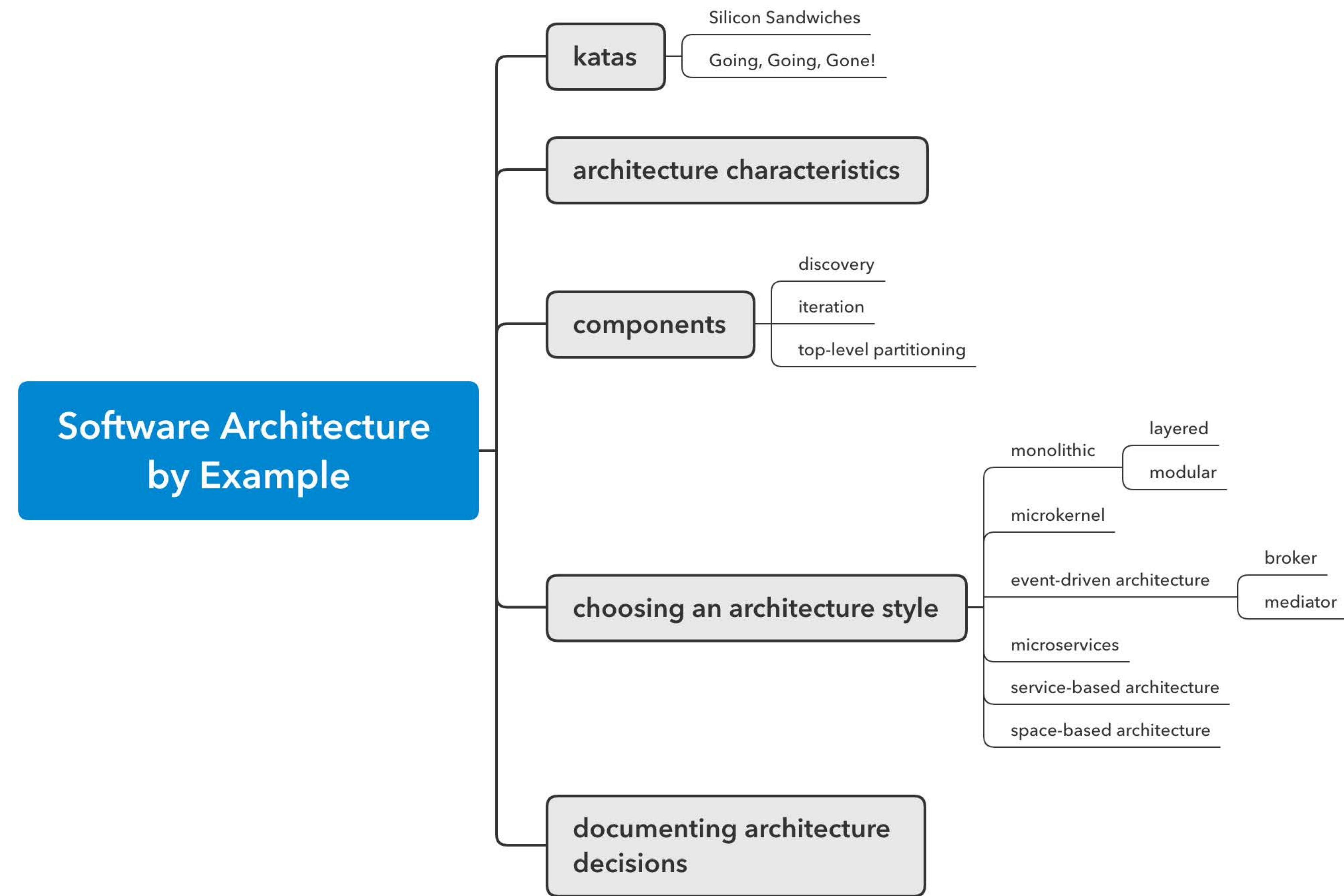
<http://www.nealford.com>

@neal4d

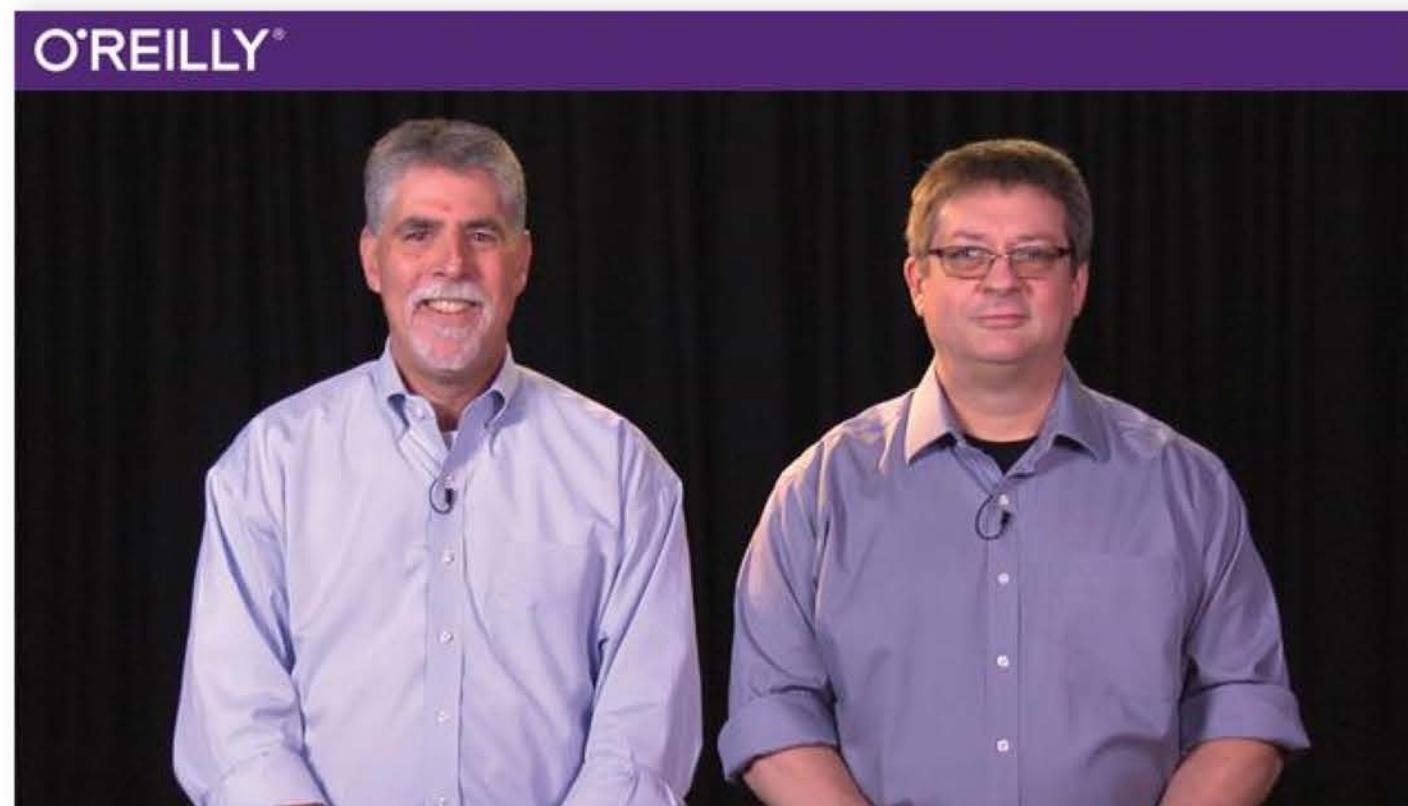


O'REILLY®





# course references



## Software Architecture Fundamentals, Second Edition

★★★★★ 9 reviews

by Mark Richards, Neal Ford

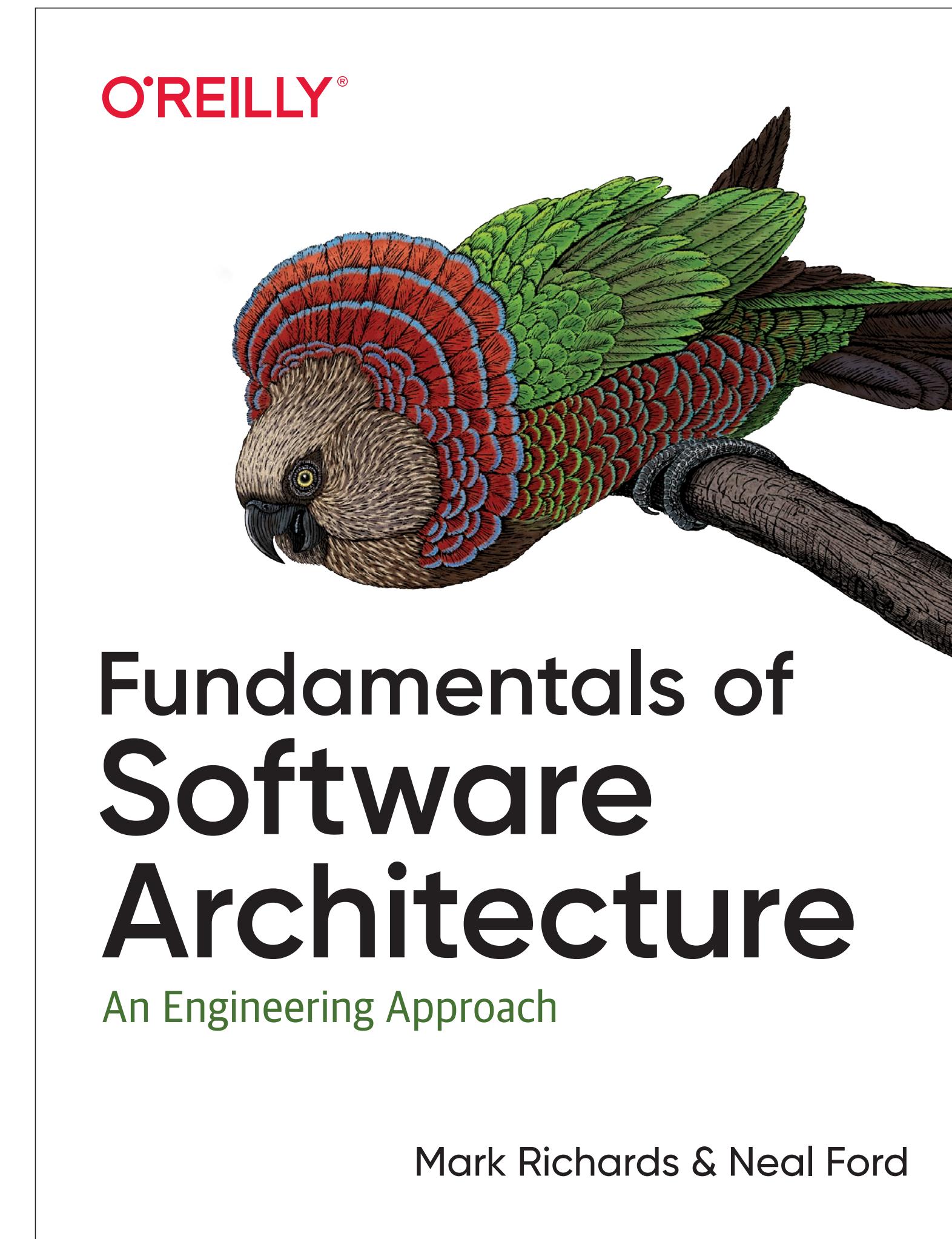
Publisher: O'Reilly Media, Inc.

*Release Date: November 2017*

ISBN: 9781491998991

<https://learning.oreilly.com/library/view/software-architecture-fundamentals/9781491998991/>

# course references



<https://learning.oreilly.com/library/view/fundamentals-of-software/9781492043447/>

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

Your Architectural Kata is...

# Silicon Sandwiches

A national sandwich shop wants to enable internet-ordering (in addition to their current call-in service)

- ***Users:*** thousands, perhaps one day millions
- ***Requirements:***
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotions/specials
  - offer local daily promotions/specials
  - accept payment online or in person/on delivery
- ***Additional Context:***
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.
  - Time to market is critical.

# First Law of Software Architecture:

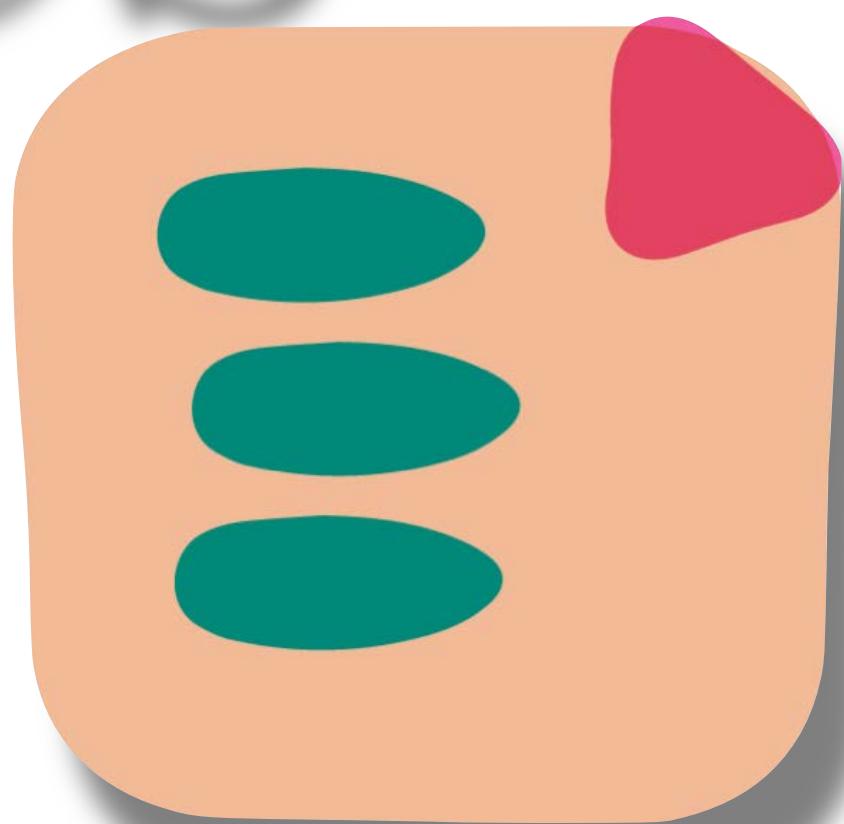
Everything in software  
architecture is a tradeoff.

First Law of Software Architecture:  
Everything in software  
architecture is a tradeoff.

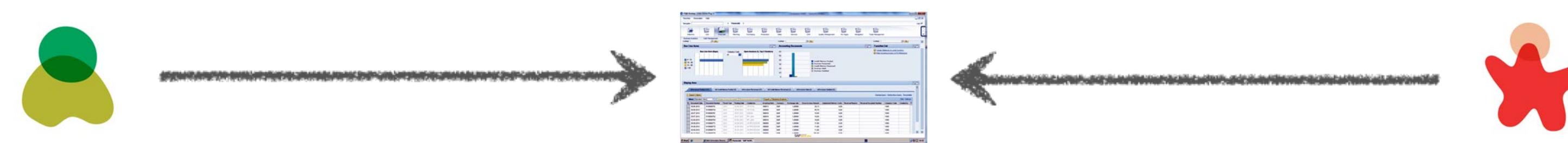
First Corollary:

If you think you've found something  
that *isn't* a tradeoff, it just means you  
haven't identified the tradeoff...yet.

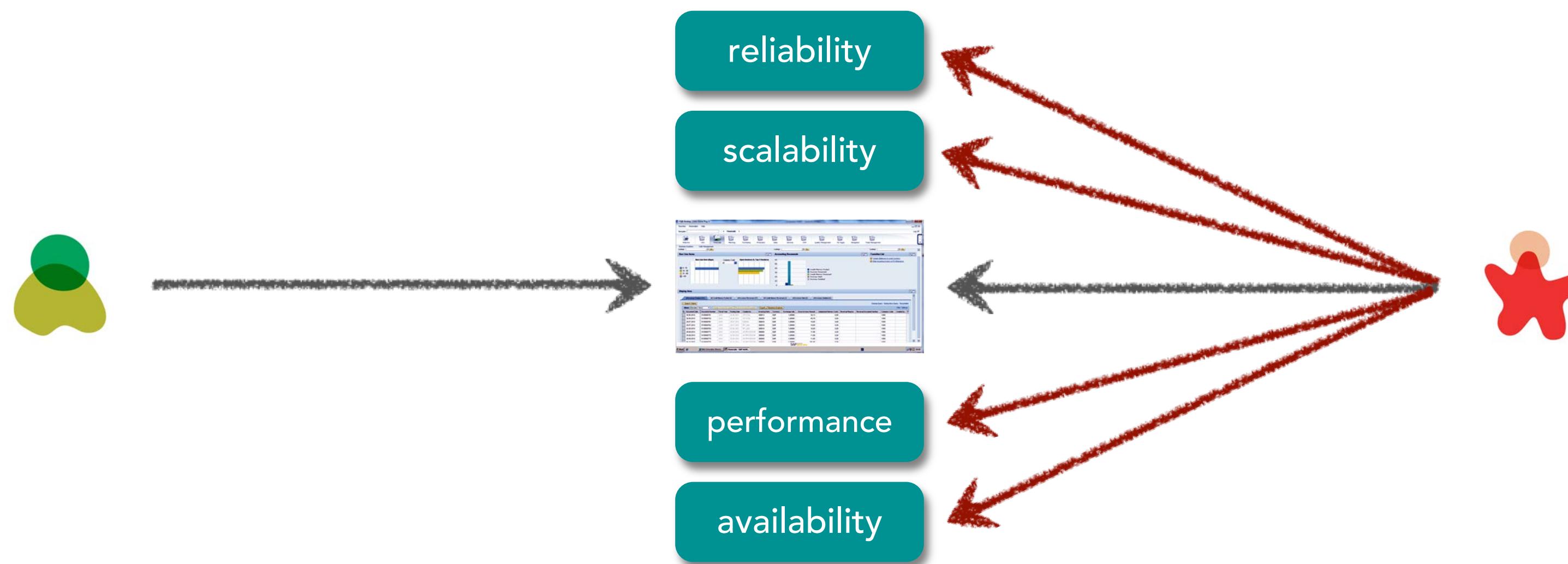
architecture  
characteristics



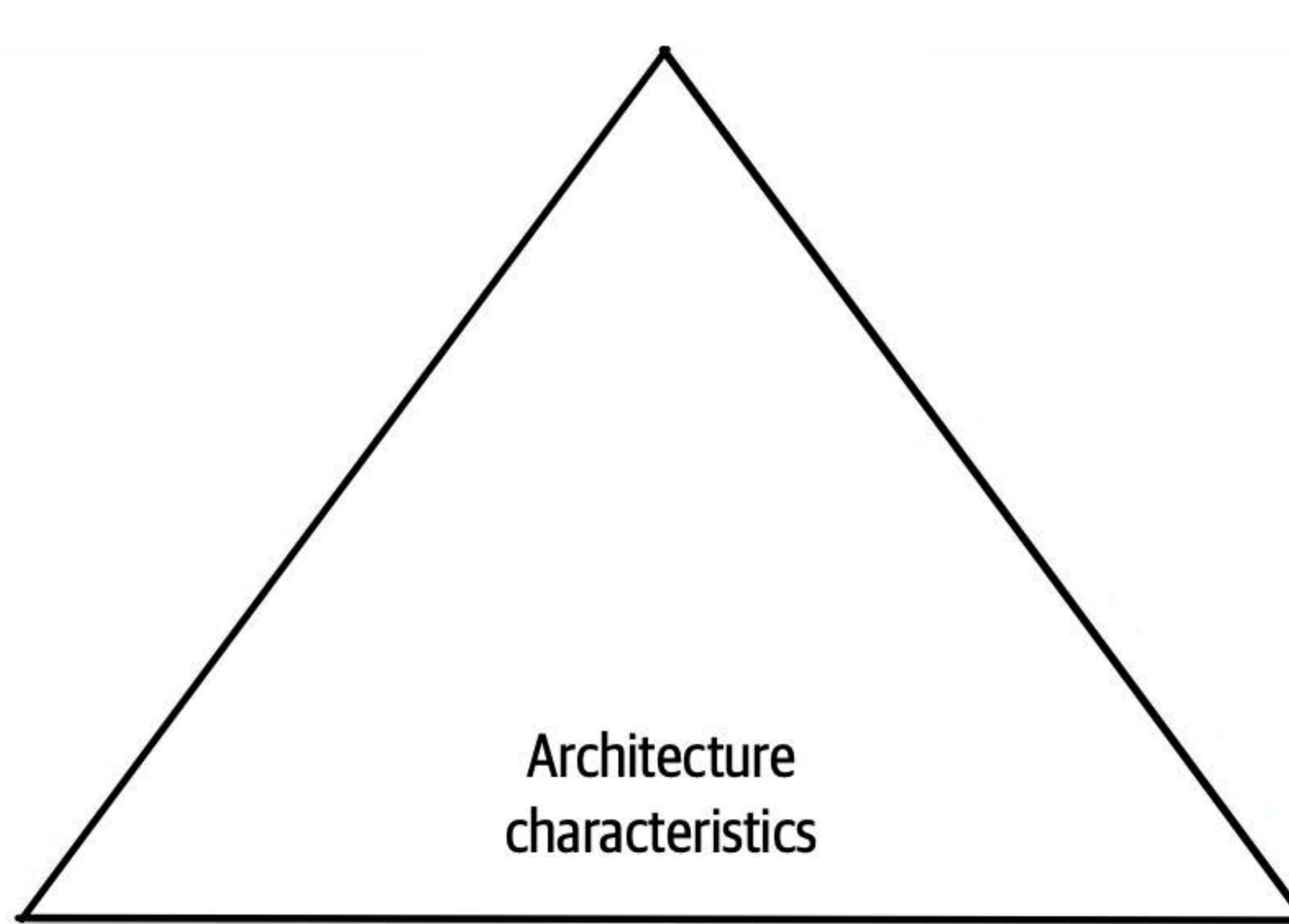
# architecture characteristics



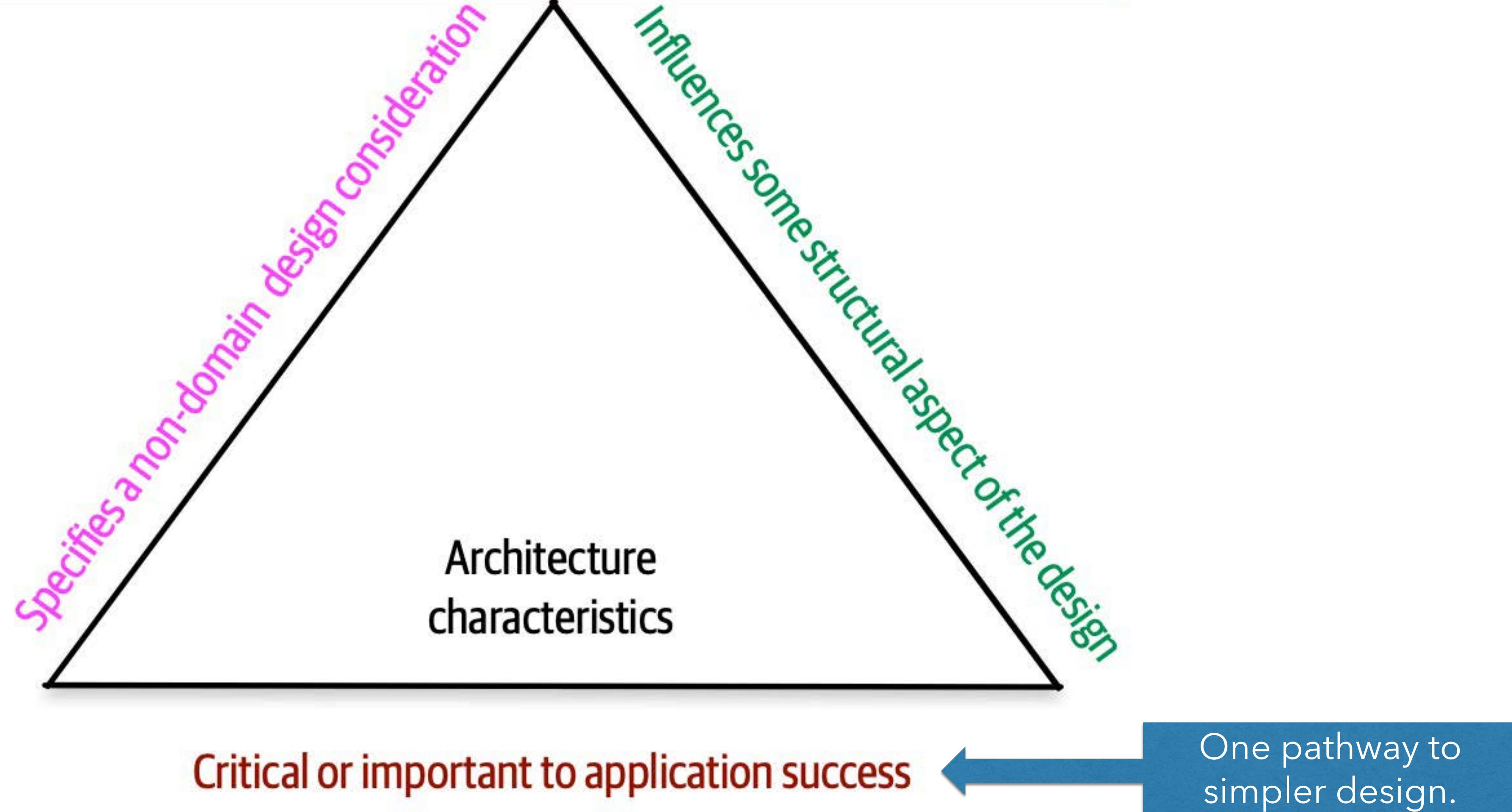
# architecture characteristics



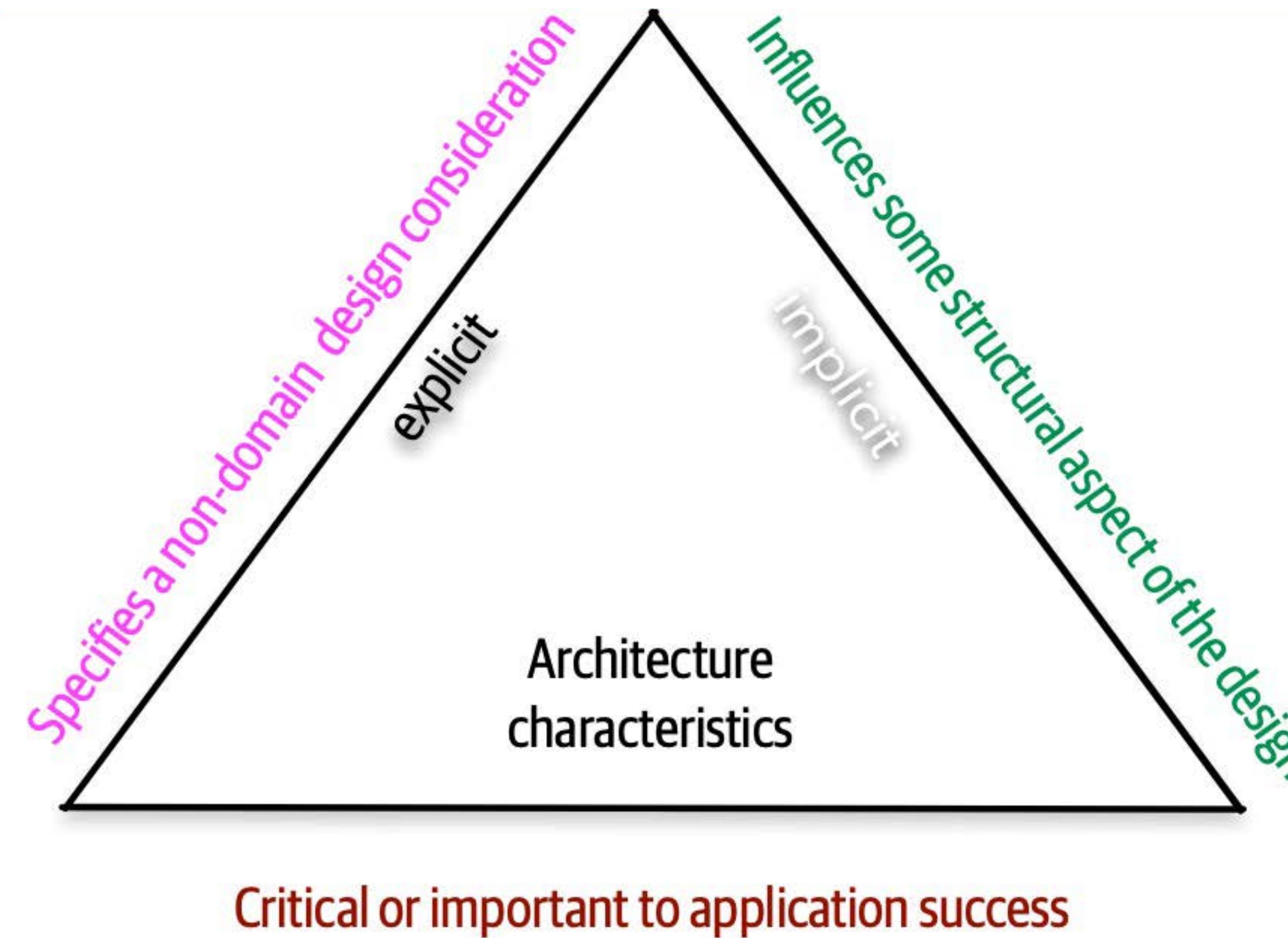
# architecture characteristics



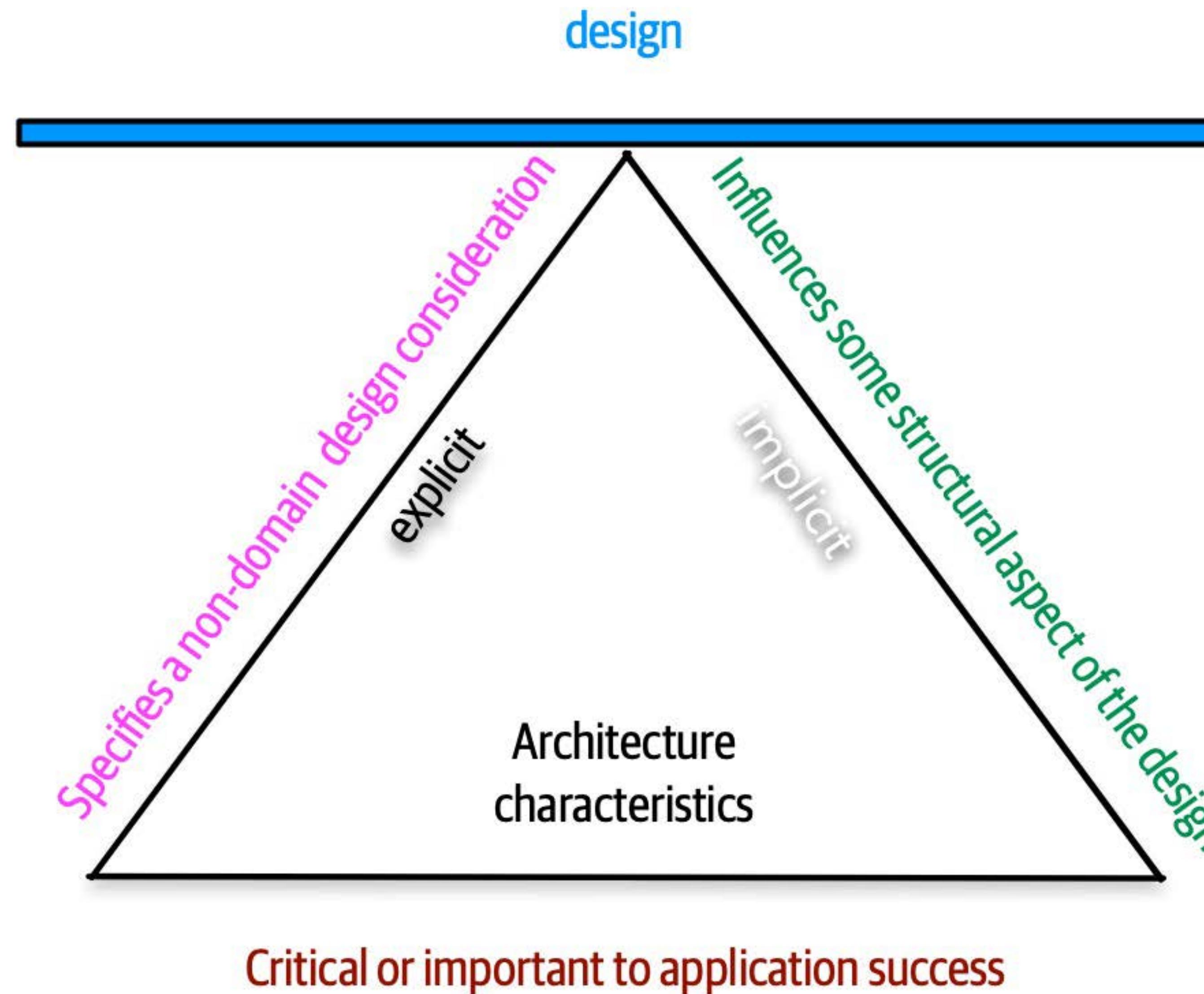
# architecture characteristics



# architecture characteristics



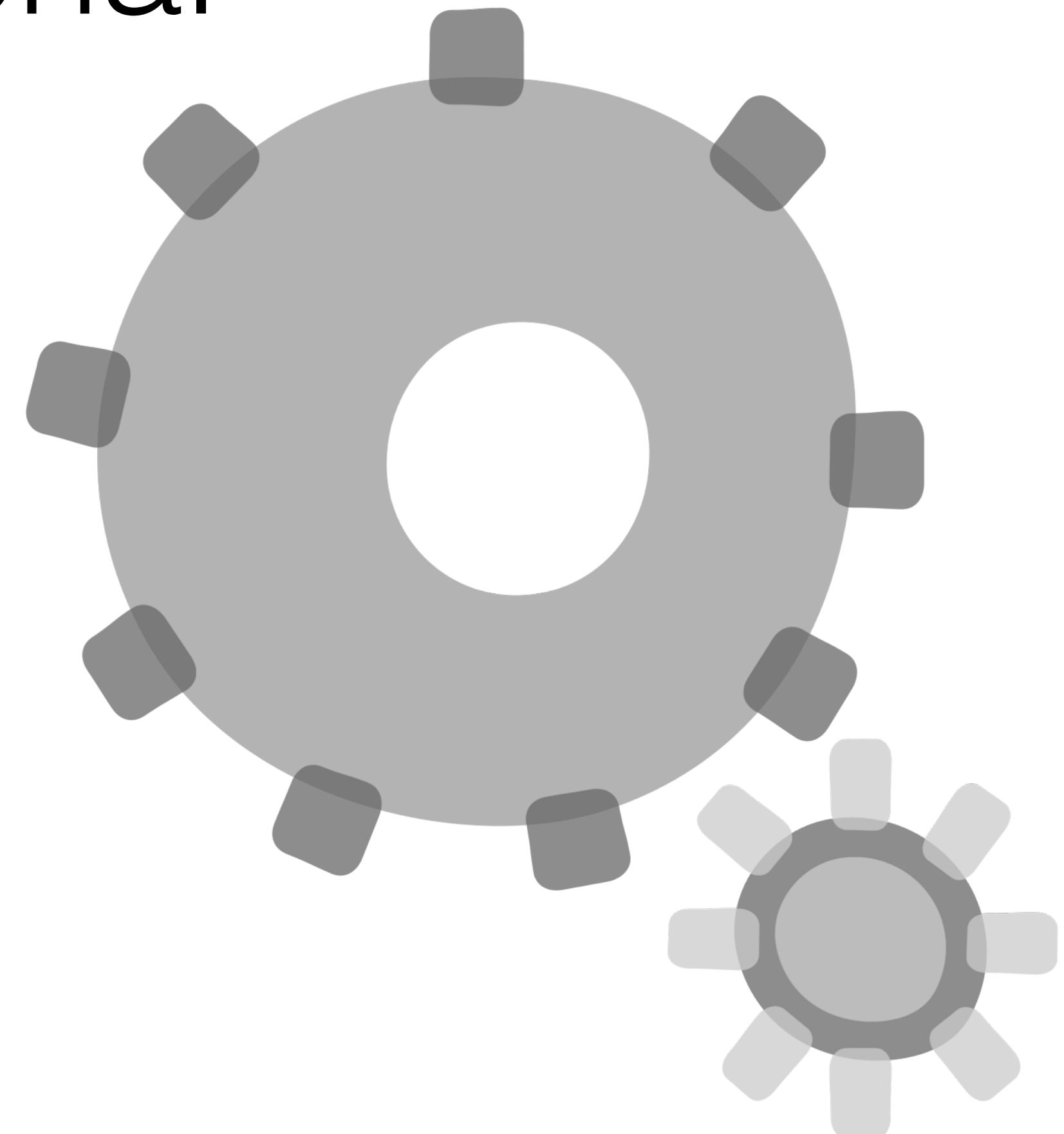
# architecture characteristics



# architecture characteristics

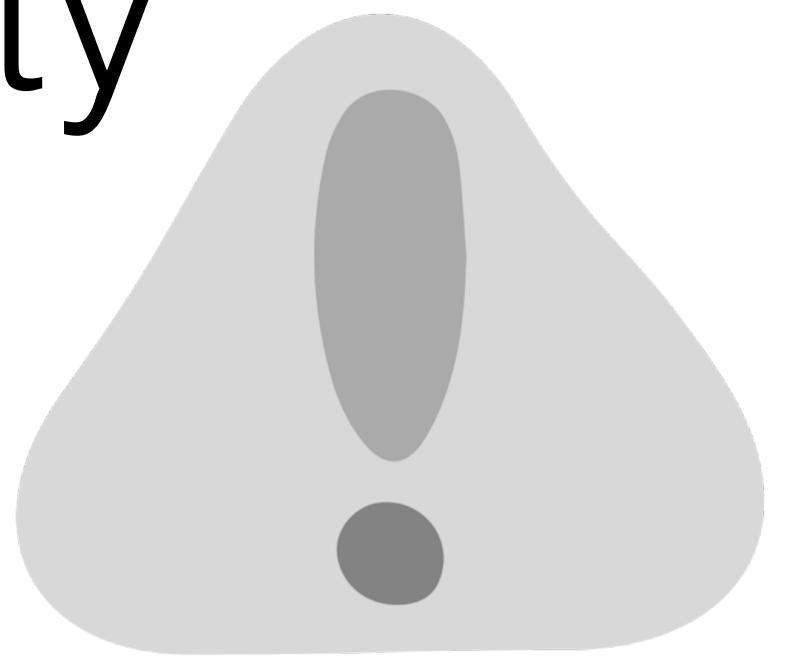
operational

Performance  
Scalability  
Elasticity  
Reliability

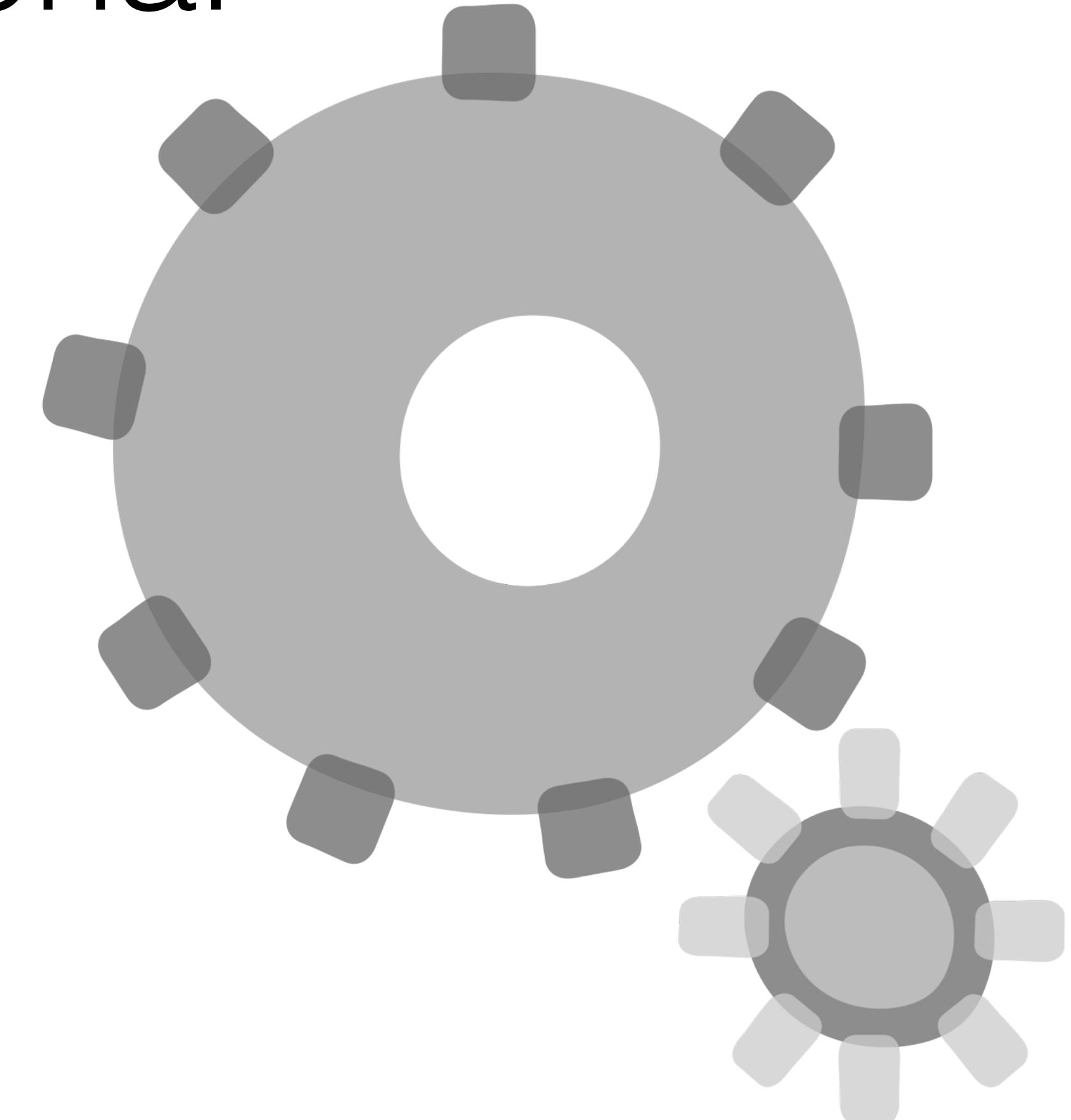


# architecture characteristics

quality

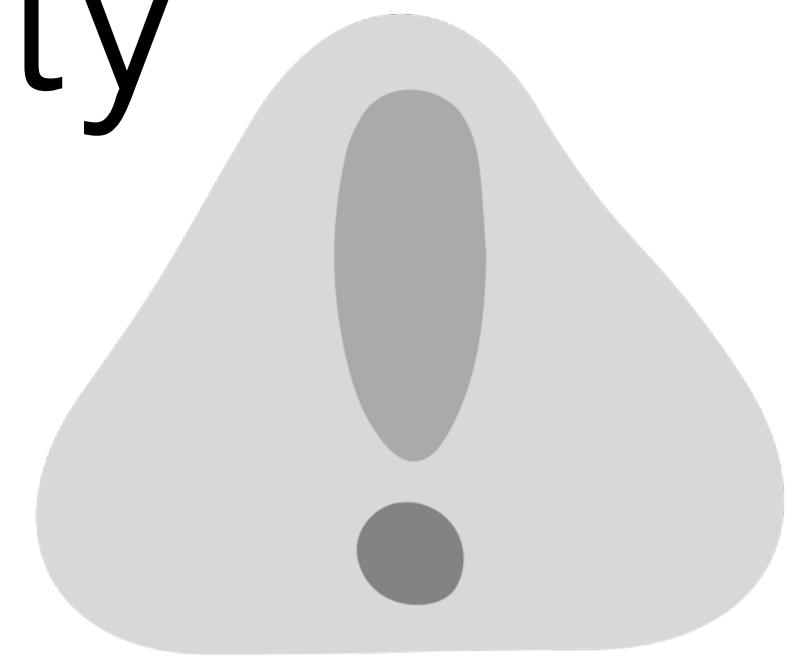


operational

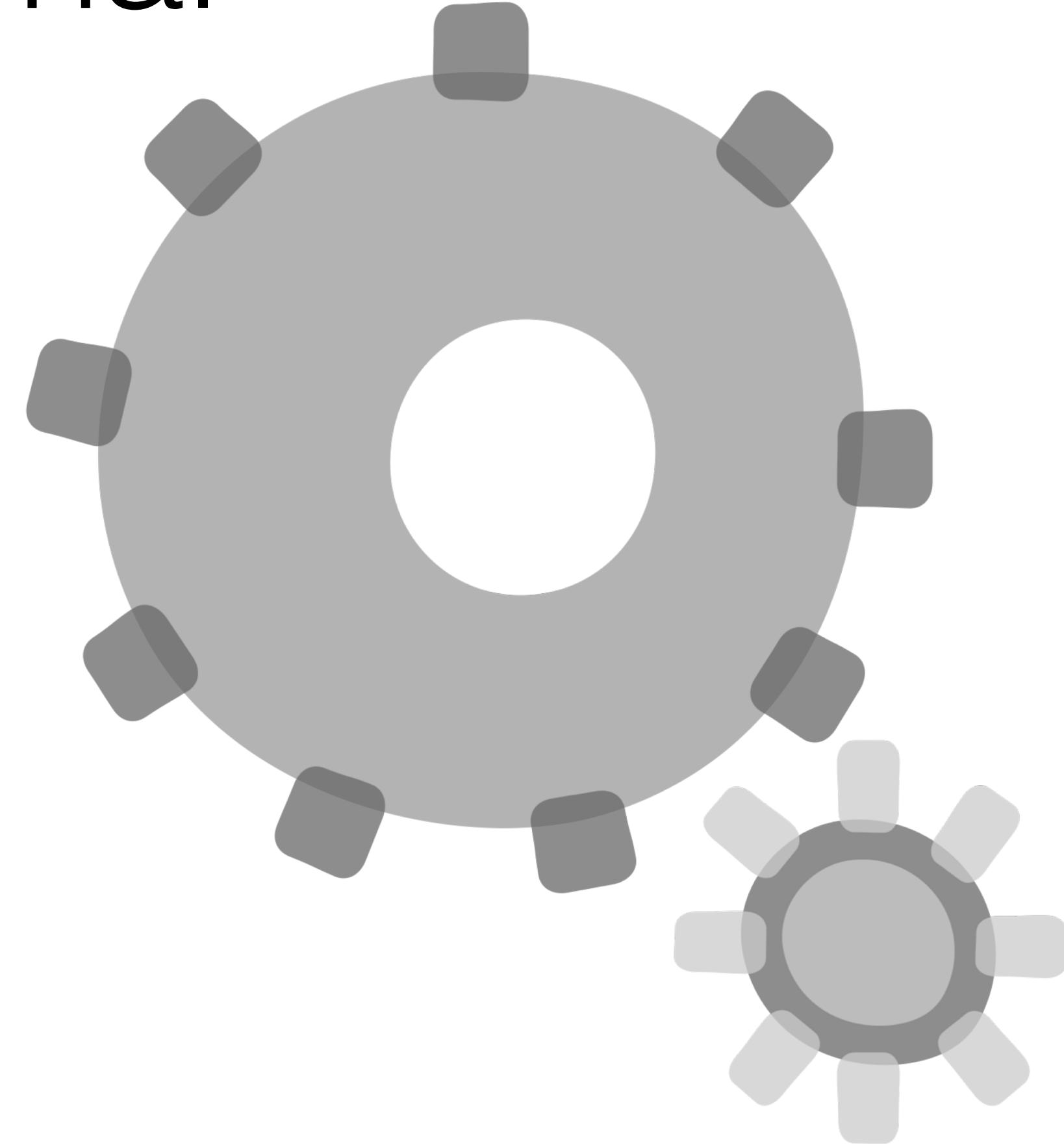


# architecture characteristics

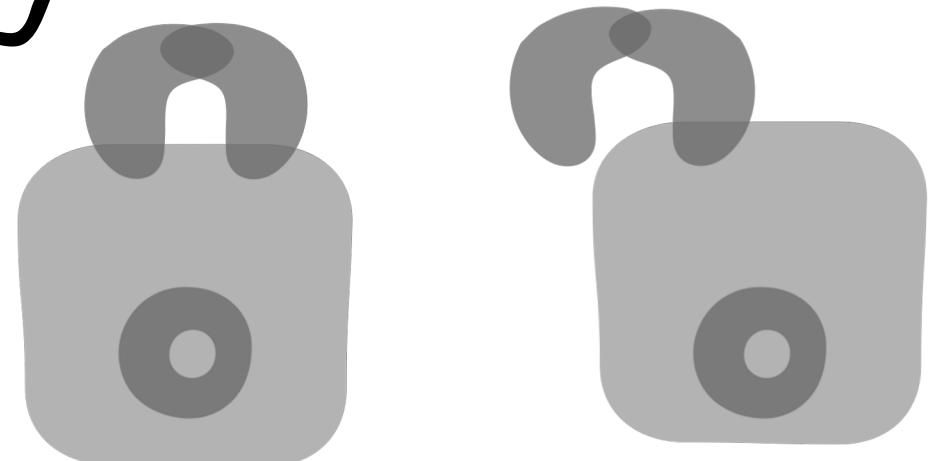
quality



operational



security



# architecture characteristics

feasibility



agility

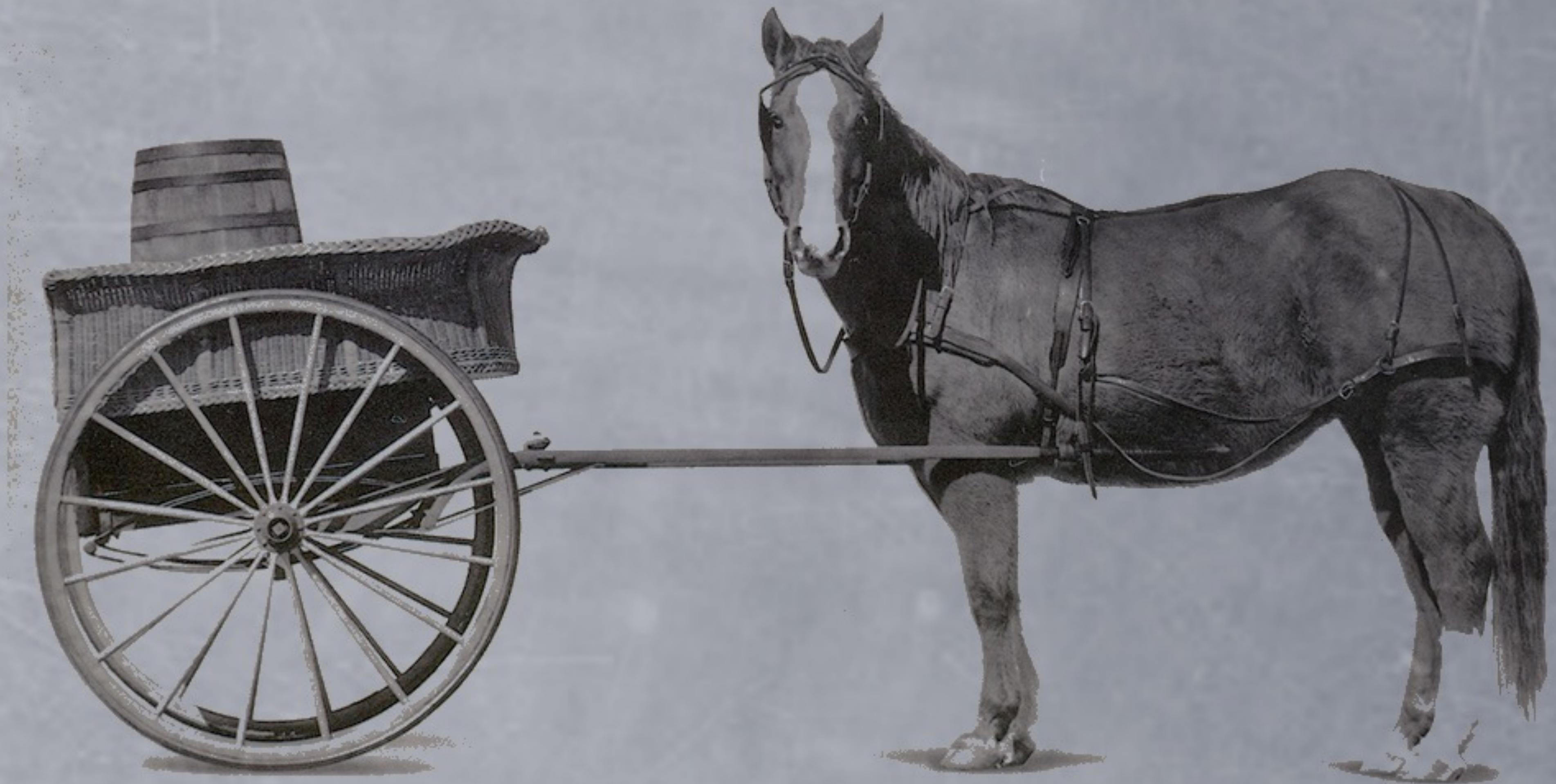


elasticity



scalability





# architecture katas

## identifying driving characteristics

Your Architectural Kata is...

### Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - video stream of the action after the fact
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

Your Architectural Kata is...

### Silicon Sandwiches

A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- **Users:** thousands, perhaps one day millions
- **Requirements:**
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotional/specials
  - offer local daily promotional/specials
  - accept payment online or in person/on delivery
- **Additional Context:**
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

Your Architectural Kata is...

# Going Going Gone!



An auction company wants to take their **auctions online** to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - **auctions must be as real-time as possible** ?
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their **auctions online** to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - **auctions must be as real-time as possible**
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

*availability   reliability   performance*

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

availability reliability performance

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

availability   reliability   performance   scalability

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

?

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

availability   reliability   performance   scalability

Your Architectural Kata is...

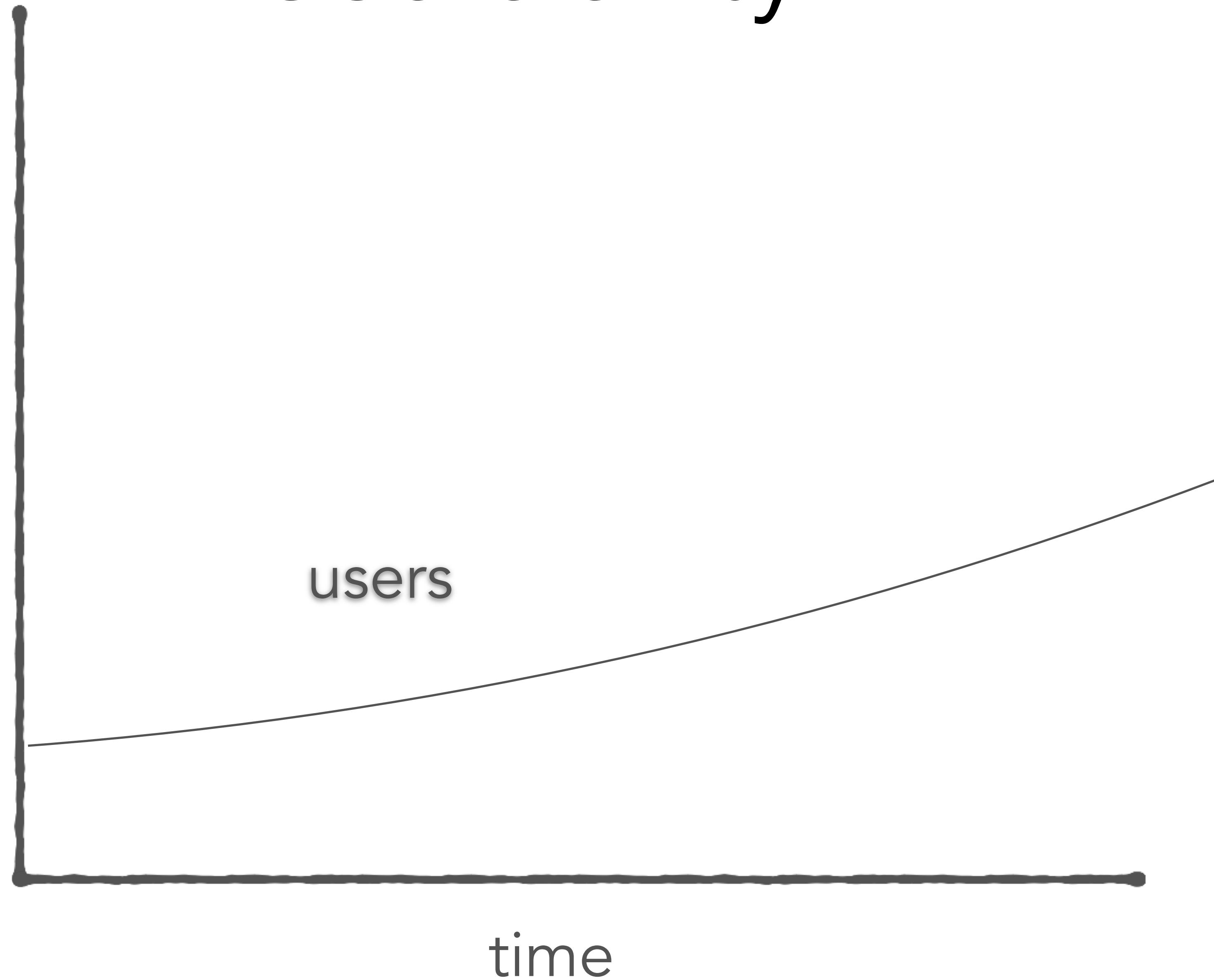
# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

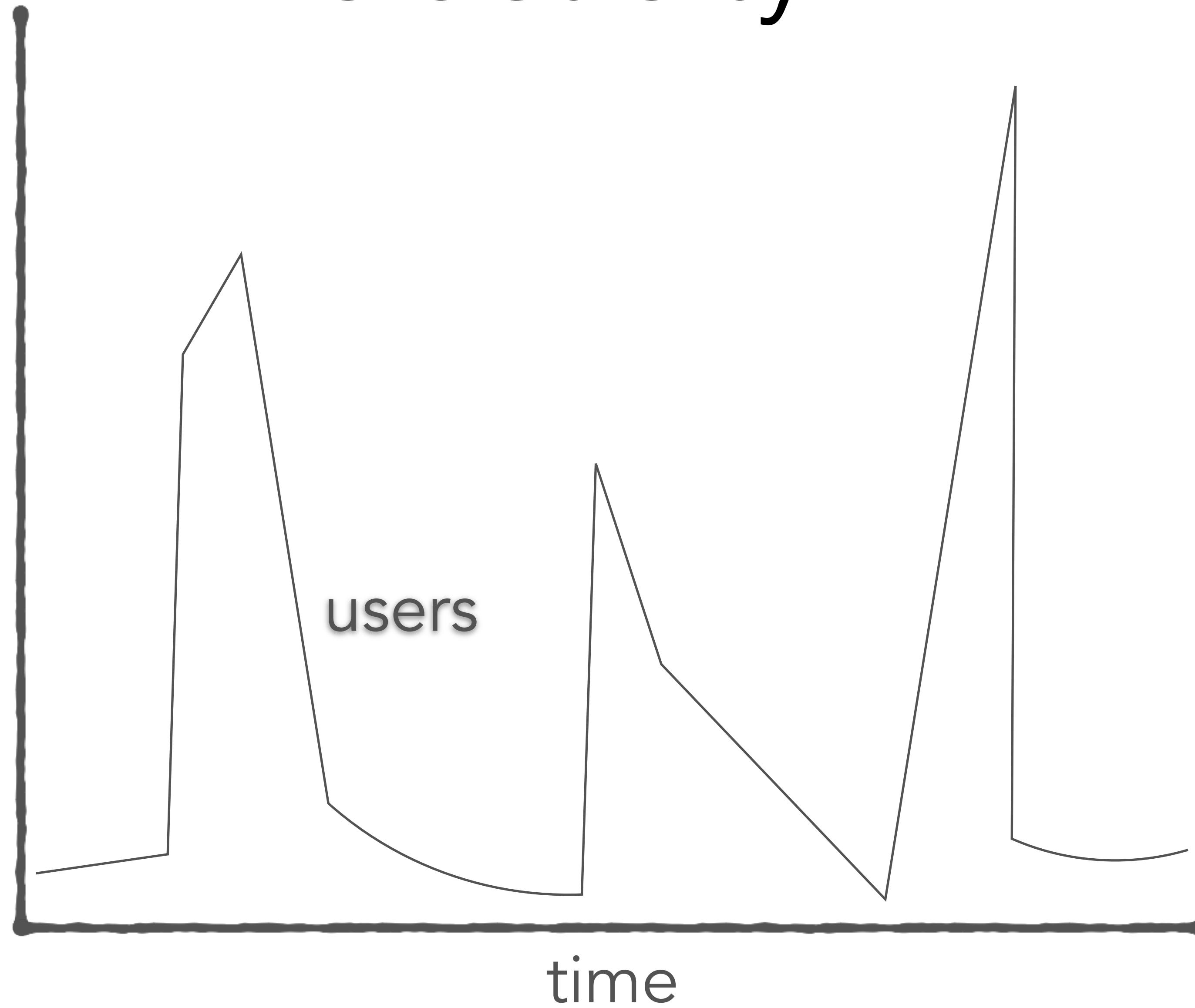
- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

availability   reliability   performance   scalability   elasticity

# scalability:



# elasticity:



Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

availability   reliability   performance   scalability   elasticity

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

?

availability reliability performance scalability elasticity

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - **bidders register with credit card; system automatically charges card if bidder wins**
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

availability reliability performance scalability elasticity (security)

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

availability   reliability   performance   scalability   elasticity   (security)

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

availability reliability performance scalability elasticity (security)

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

concurrency

availability reliability performance scalability elasticity (security)

Your Architectural Kata is...

# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

concurrency

availability reliability performance scalability elasticity (security)

Your Architectural Kata is...

# Silicon Sandwiches

A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- ***Users:*** thousands, perhaps one day millions
- ***Requirements:***
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotions/specials
  - offer local daily promotions/specials
  - accept payment online or in person/on delivery
- ***Additional Context:***
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.
  - Time to market is critical.

Your Architectural Kata is...

# Silicon Sandwiches

A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- **Users:** thousands, perhaps one day millions
- **Requirements:**
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotions/specials
  - offer local daily promotions/specials
  - accept payment online or in person/on delivery
- **Additional Context:**
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.
  - Time to market is critical.

performance availability reliability

scalability elasticity i18n l10n

Your Architectural Kata is...

# Silicon Sandwiches

A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- **Users:** thousands, perhaps one day millions
- **Requirements:**
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotions/specials
  - offer local daily promotions/specials
  - accept payment online or in person/on delivery

performance availability reliability  
scalability elasticity i18n l10n
- **Additional Context:**
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.
  - Time to market is critical.

Global Local  
Customizability Customizability

Your Architectural Kata is...

# Silicon Sandwiches

A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- **Users:** thousands, perhaps one day millions
- **Requirements:**
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotions/specials
  - offer local daily promotions/specials
  - accept payment online or in person/on delivery
- **Additional Context:**
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.
  - Time to market is critical.

performance availability reliability  
scalability elasticity  
Customizability {  
location  
sales  
recipe

Your Architectural Kata is...

# Silicon Sandwiches

A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- **Users:** thousands, perhaps one day millions
- **Requirements:**
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotions/specials
  - offer local daily promotions/specials
  - accept payment online or in person/on delivery
- **Additional Context:**
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.
  - Time to market is critical.



# architecture katas

## identifying major components

Your Architectural Kata is...

### Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - video stream of the action after the fact
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

Your Architectural Kata is...

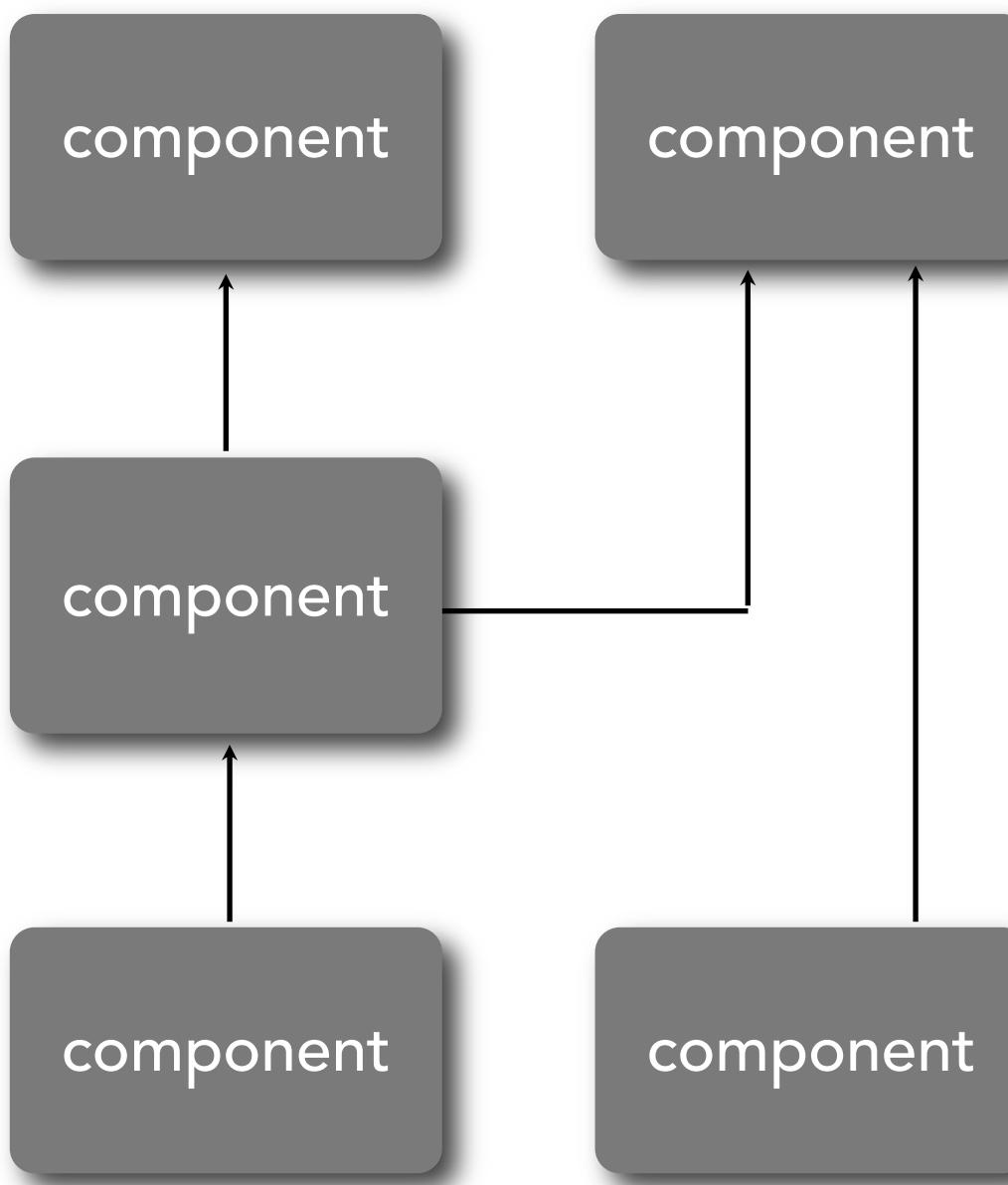
### Silicon Sandwiches

A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- **Users:** thousands, perhaps one day millions
- **Requirements:**
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotional/specials
  - offer local daily promotional/specials
  - accept payment online or in person/on delivery
- **Additional Context:**
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.

# component identification

as an architect, you should think about the artifacts within the architecture in terms of *components*



## **component:**

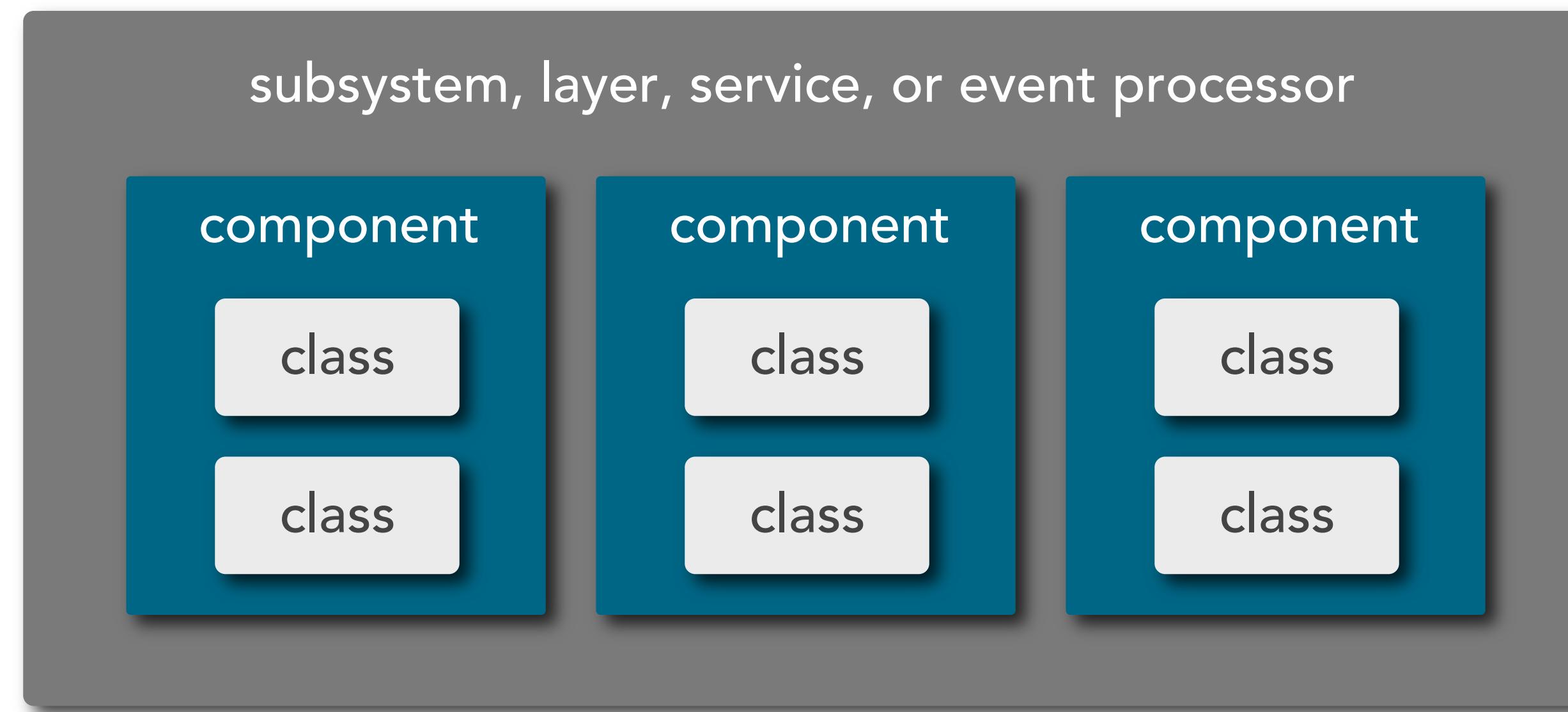
building block of the application

well defined set of operations

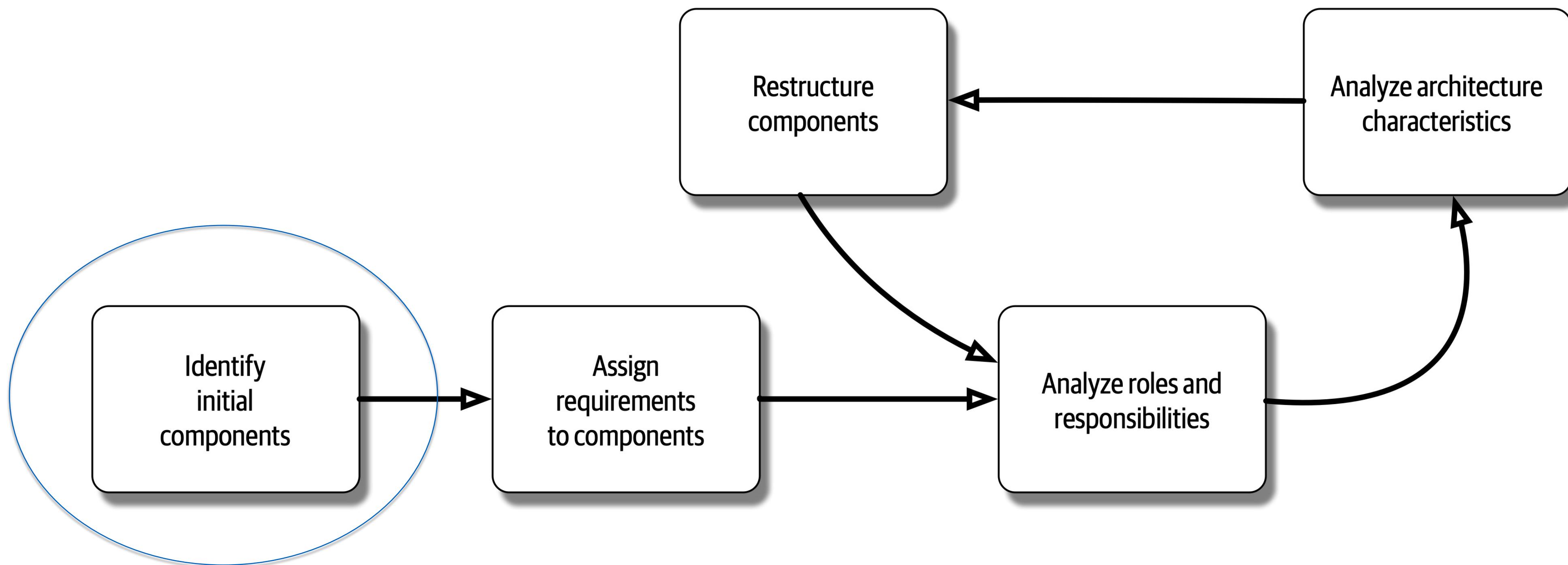
well defined role and responsibility

# component identification

## component scope

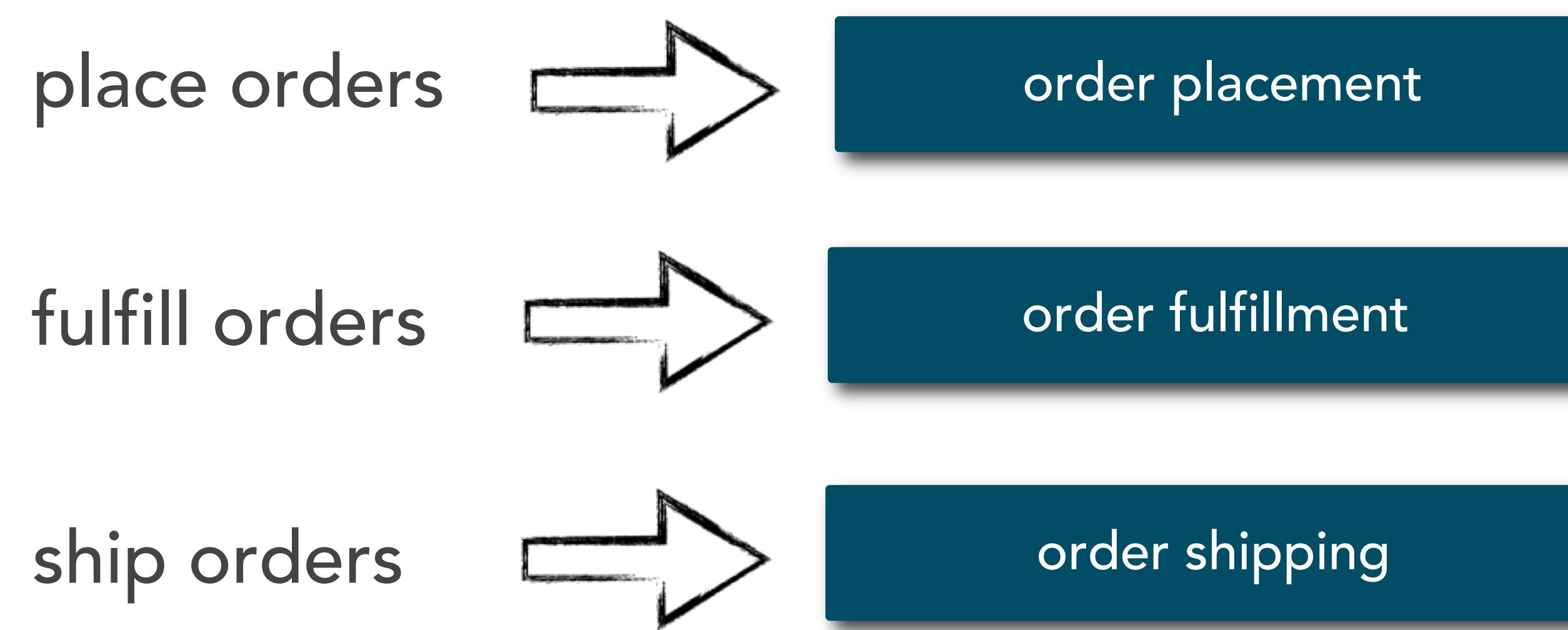


# component identification

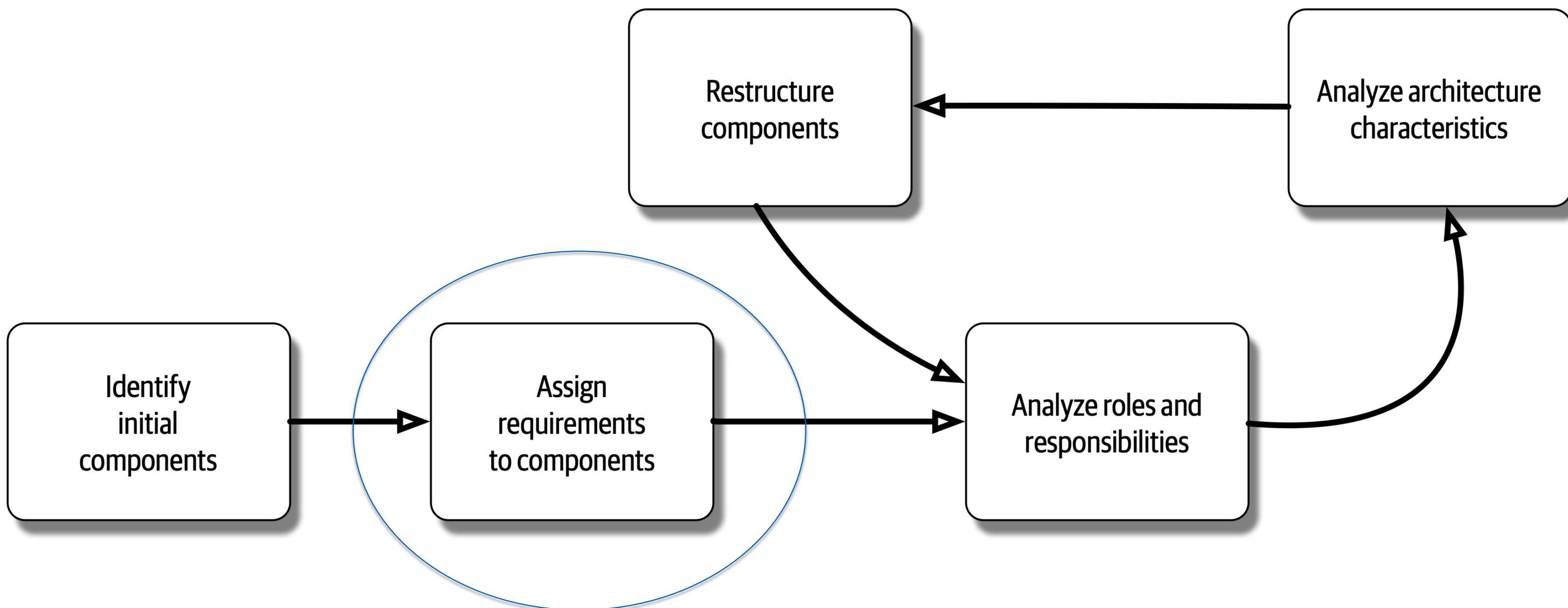


# component identification

identify initial components using core functionality

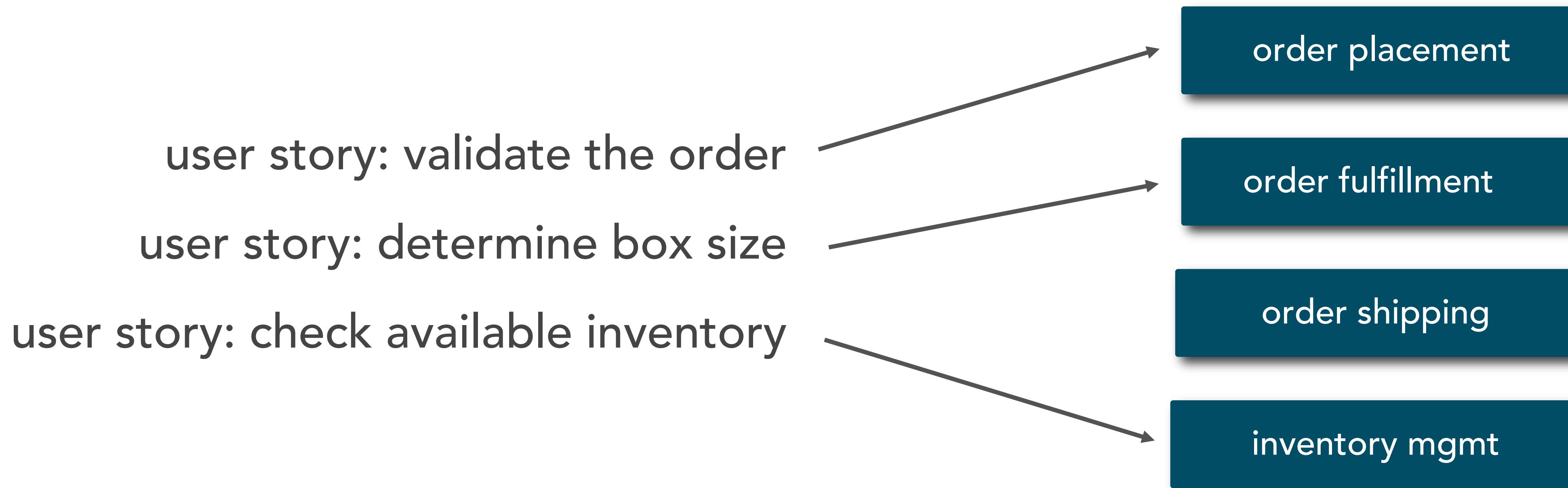


# component identification

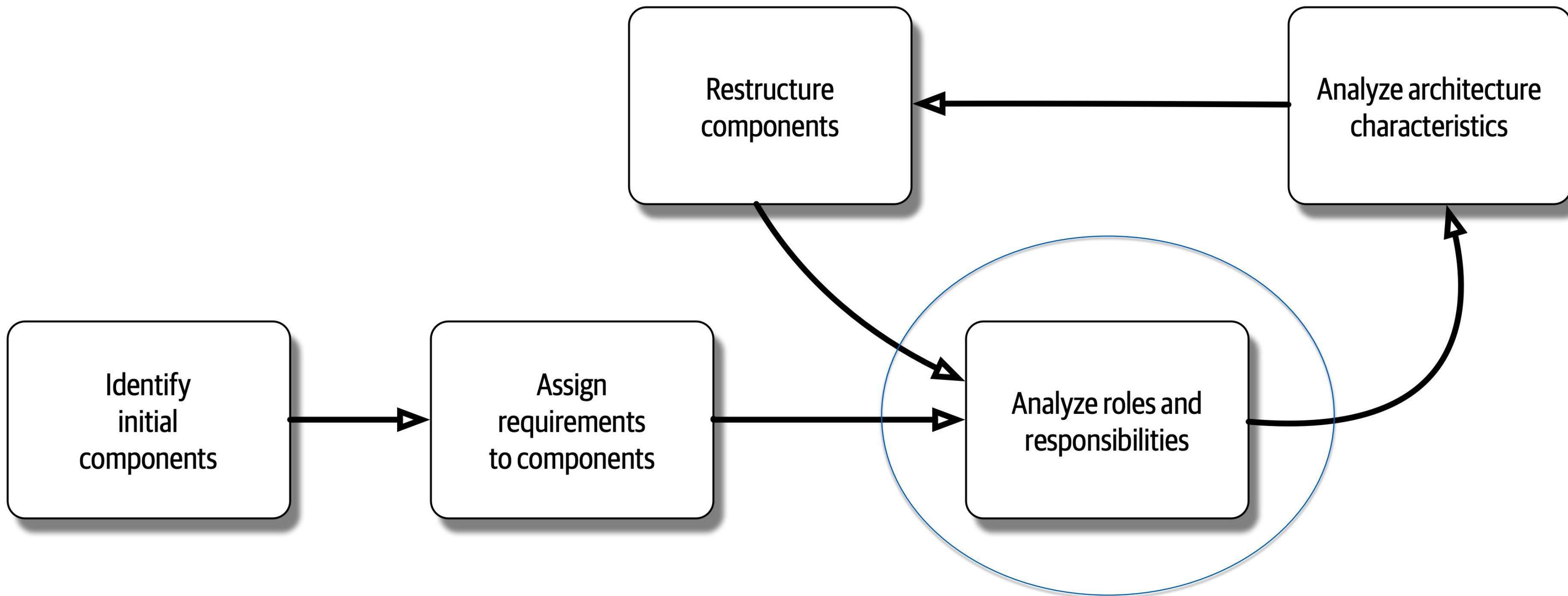


# component identification

assign requirements, use cases, or user stories to a component



# component identification



# component identification

## component granularity



order manager

responsible for creating, deleting, and updating orders. also responsible for shipping the order and tracking the order once it has been shipped. this component is also responsible for notifying the customer each time the order status changes.

# component identification

## component granularity

order maintenance

responsible for creating, deleting, and updating orders.

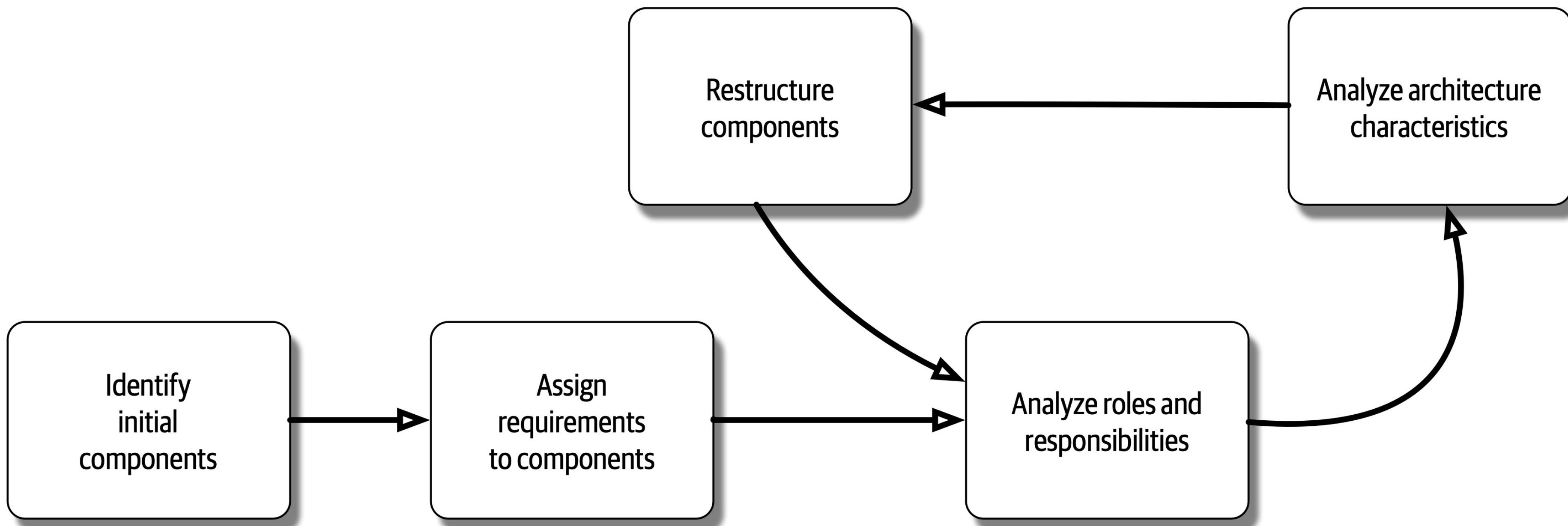
order shipment

responsible for shipping and tracking orders

customer notification

responsible for notifying the customer when the order status changes.

# component identification



Your Architectural Kata is...

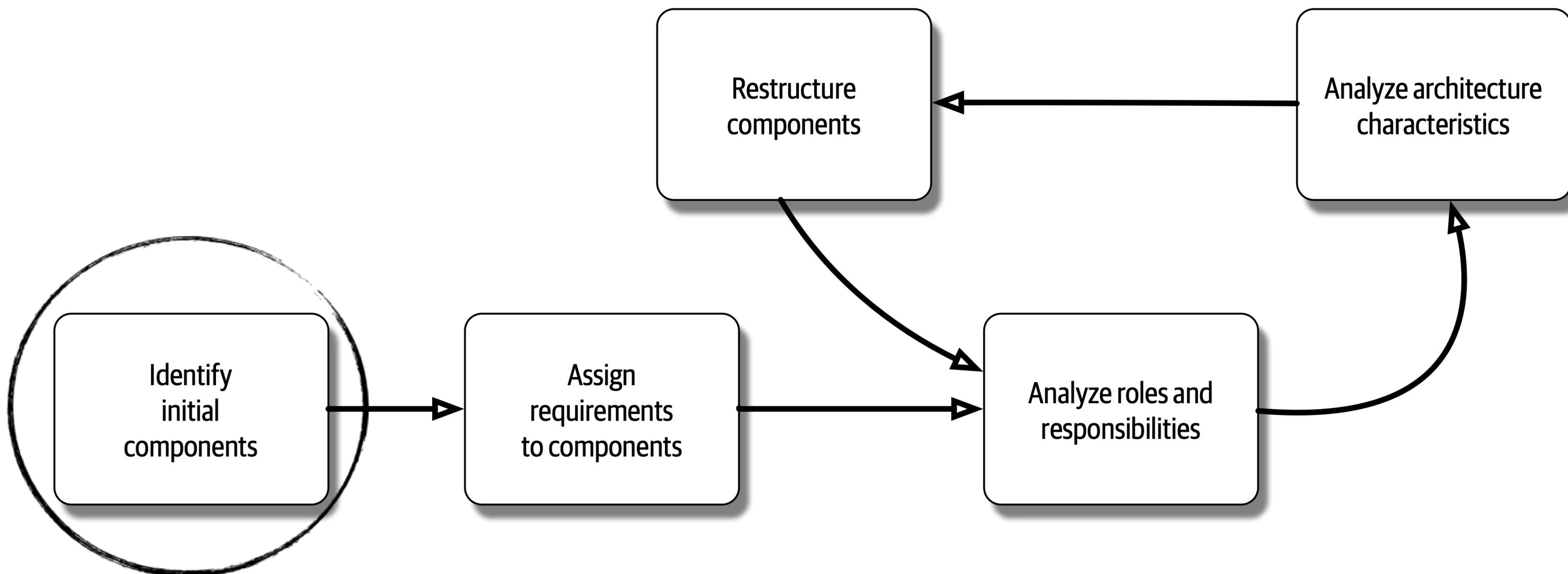
# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

Your Architectural Kata is...

# Going Going Gone!



Your Architectural Kata is...

# Going Going Gone!

the “entity trap”

auctions

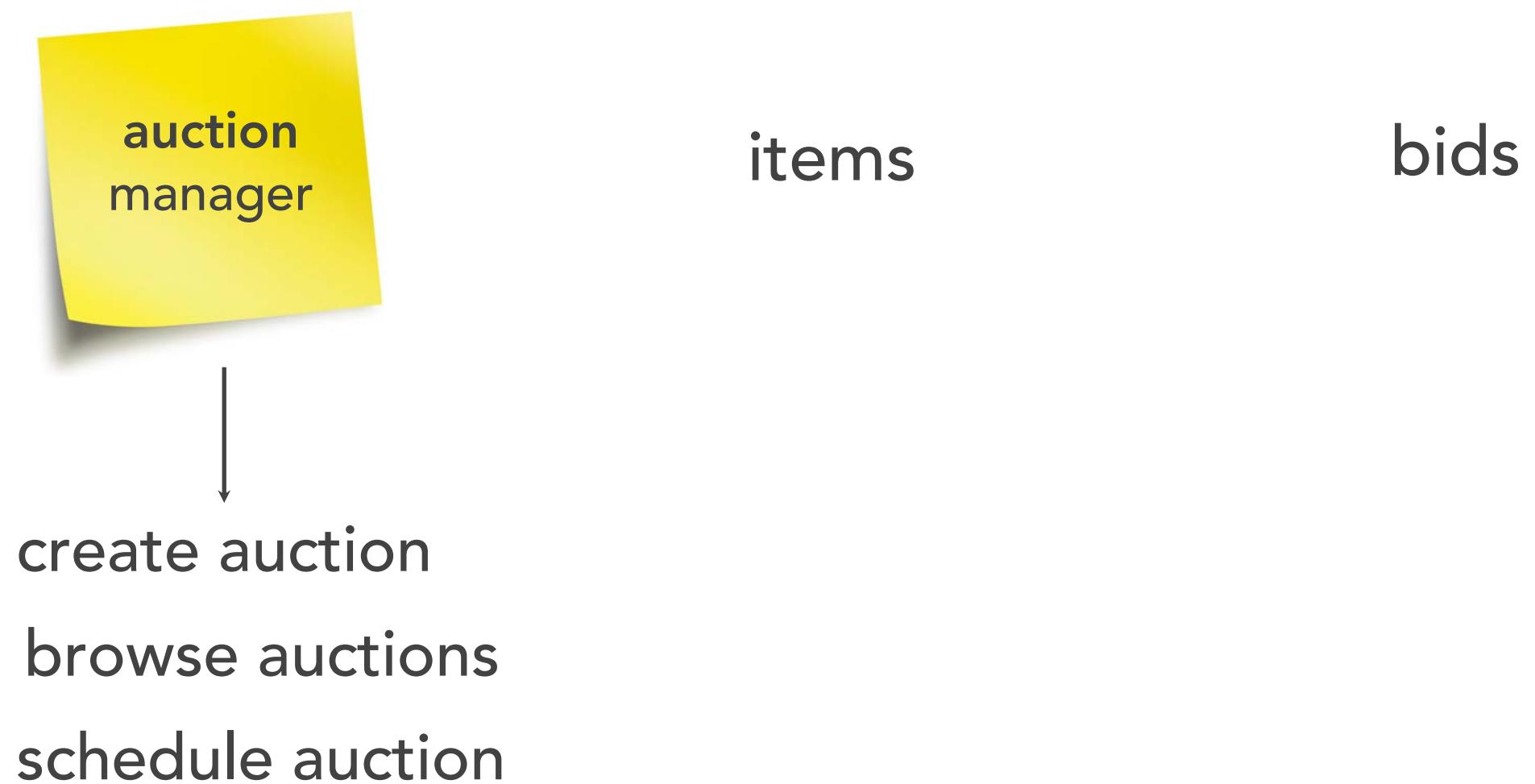
items

bids

Your Architectural Kata is...

# Going Going Gone!

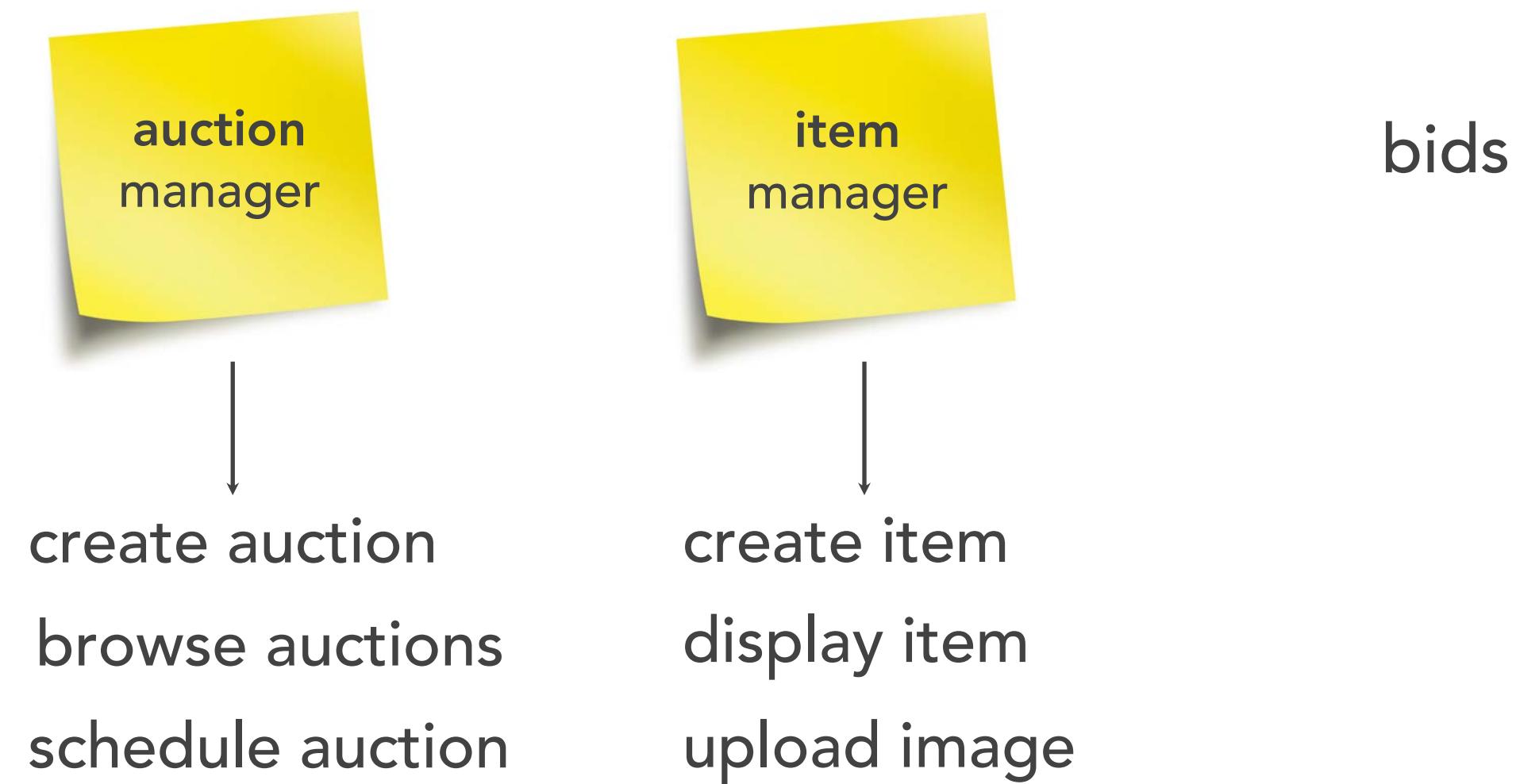
the “entity trap”



Your Architectural Kata is...

# Going Going Gone!

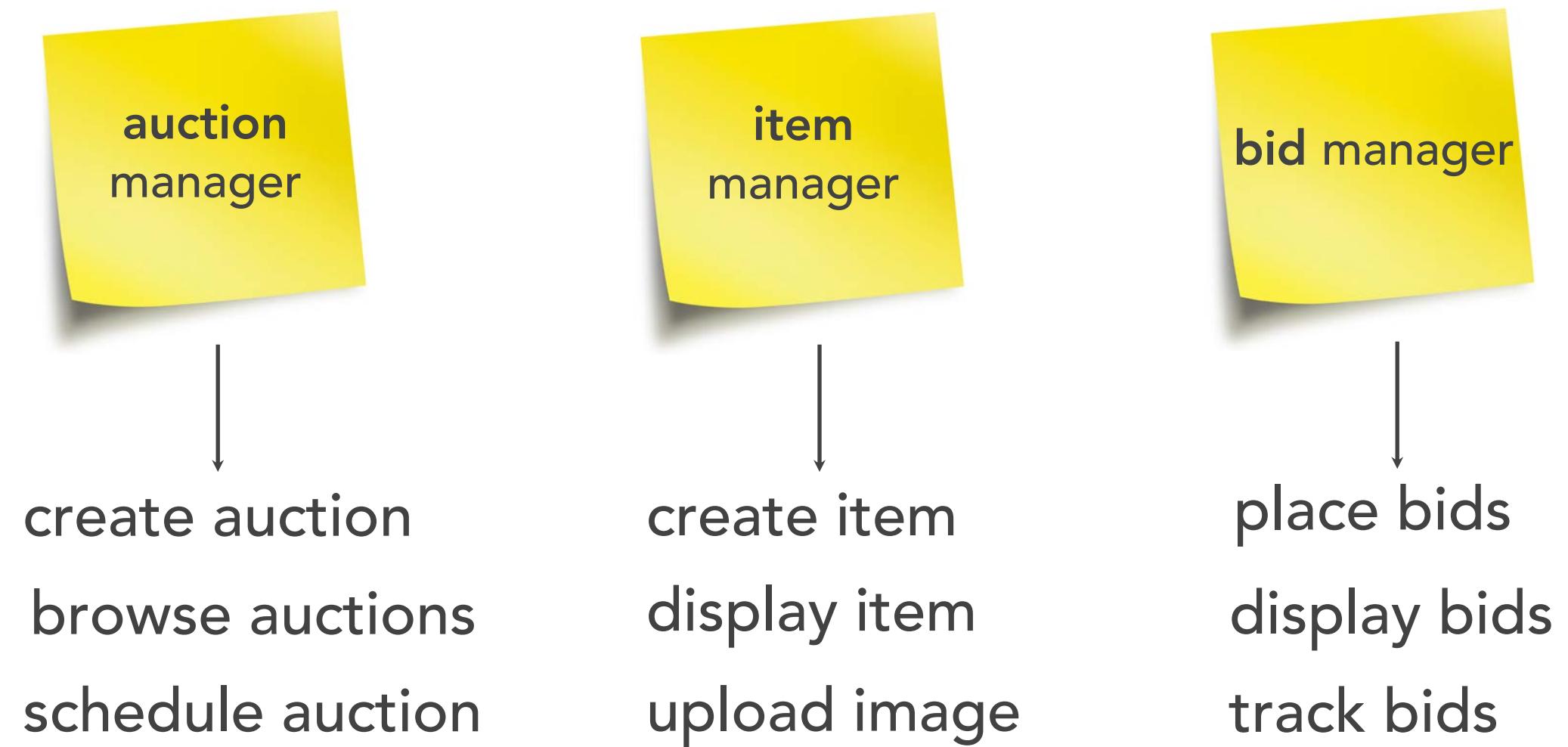
the “entity trap”



Your Architectural Kata is...

# Going Going Gone!

the “entity trap”



Your Architectural Kata is...

# Going Going Gone!

workflow approach

create auction —> find auction —> sign up —> watch auction —> place bid

Your Architectural Kata is...

# Going Going Gone!

## workflow approach

create auction —→ find auction —→ sign up —→ watch auction —→ place bid



Your Architectural Kata is...

# Going Going Gone!

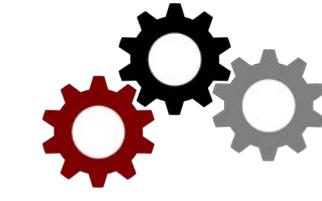
actor/action approach



bidder



auctioneer



system

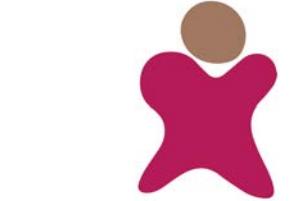
Your Architectural Kata is...

# Going Going Gone!

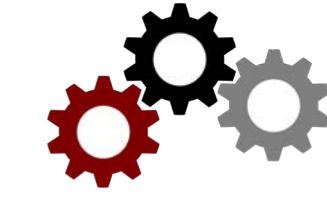
actor/action approach



bidder



auctioneer



system



view live video stream

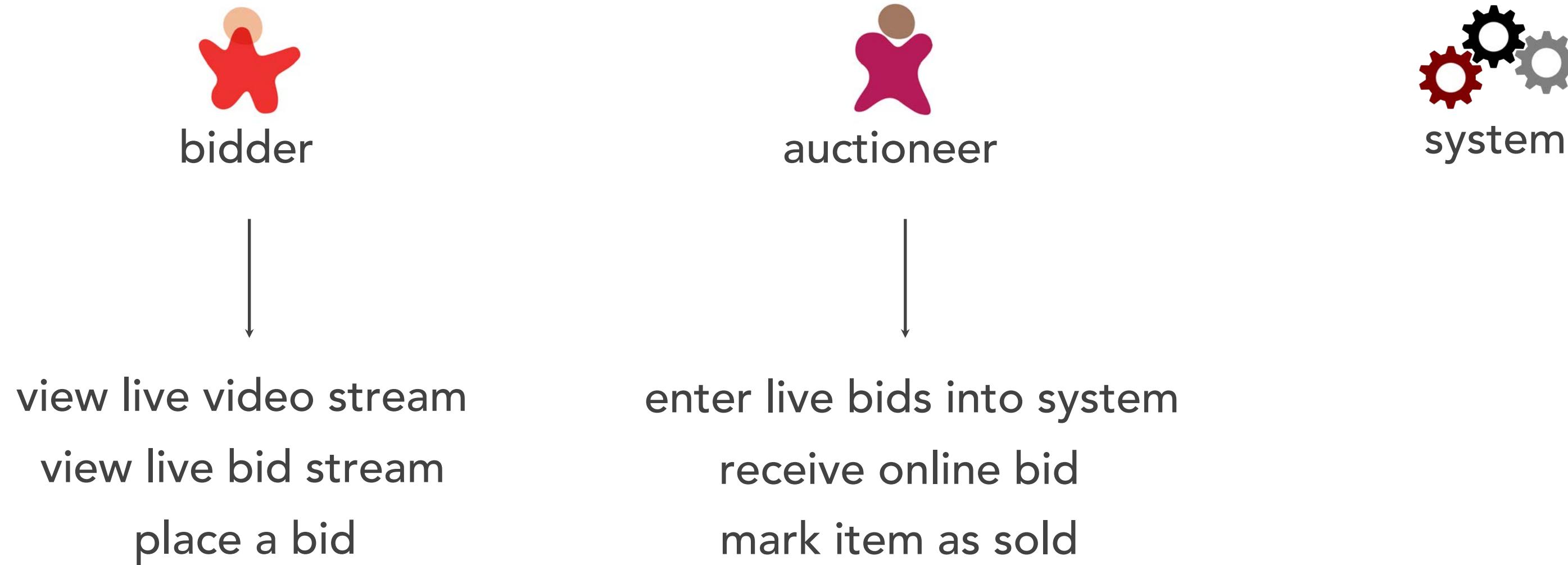
view live bid stream

place a bid

Your Architectural Kata is...

# Going Going Gone!

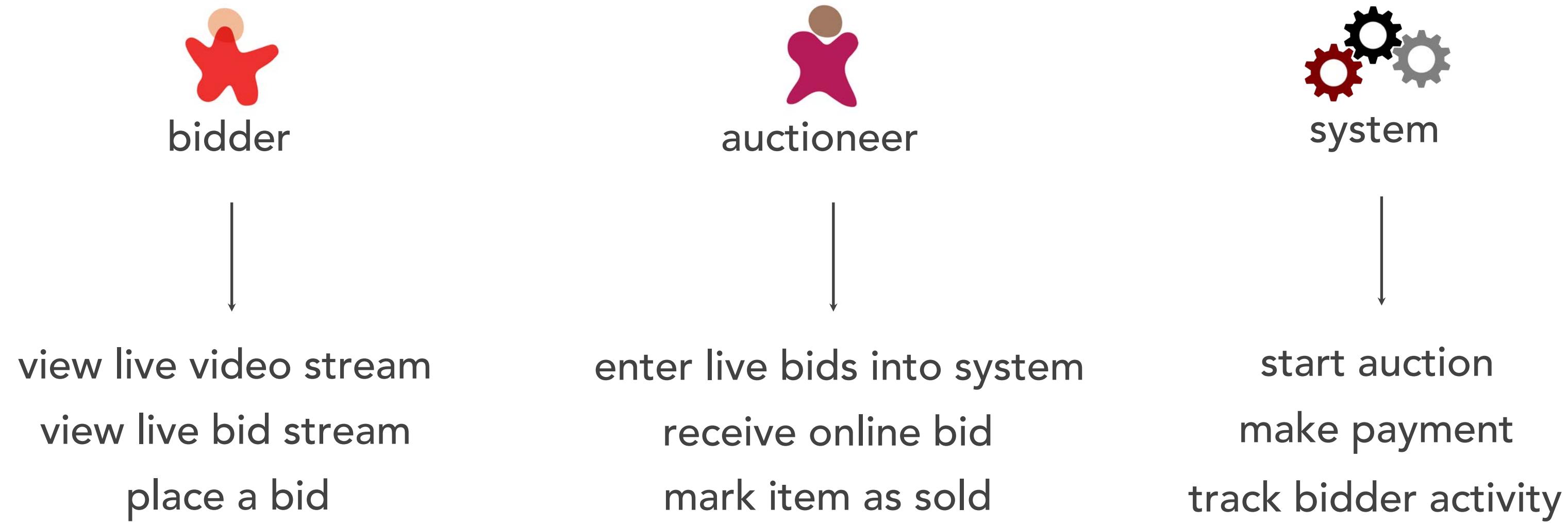
actor/action approach



Your Architectural Kata is...

# Going Going Gone!

actor/action approach



Your Architectural Kata is...

# Going Going Gone!



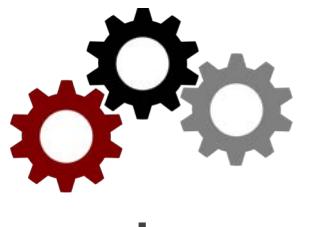
bidder

- view live video stream
- view live bid stream
- place a bid



auctioneer

- receive online bid
- enter live bids into system
- mark item as sold



system

- start auction
- make payment
- track bidder activity

Your Architectural Kata is...

# Going Going Gone!



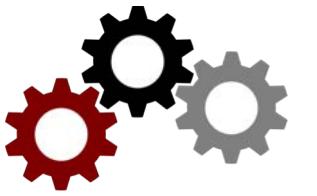
bidder

- view live video stream
- view live bid stream
- place a bid



auctioneer

- receive online bid
- enter live bids into system
- mark item as sold



system

- start auction
- make payment
- track bidder activity

Your Architectural Kata is...

# Going Going Gone!



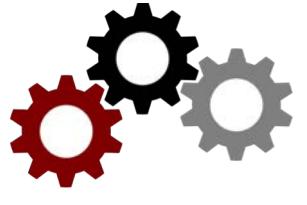
bidder

- view live video stream
- view live bid stream
- place a bid



auctioneer

- receive online bid
- enter live bids into system
- mark item as sold



system

- start auction
- make payment
- track bidder activity



auction  
session



Your Architectural Kata is...

# Going Going Gone!



bidder

- view live video stream
- view live bid stream
- place a bid



auctioneer

- receive online bid
- enter live bids into system
- mark item as sold



system

- ✓ start auction
- make payment
- track bidder activity



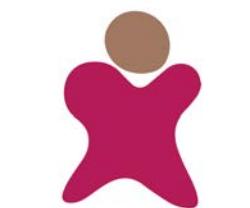
Your Architectural Kata is...

# Going Going Gone!



bidder

- view live video stream
- view live bid stream
- place a bid



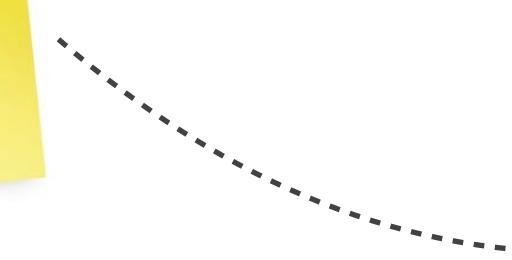
auctioneer

- receive online bid
- enter live bids into system
- mark item as sold



system

- ✓ start auction
- make payment
- track bidder activity



Your Architectural Kata is...

# Going Going Gone!



bidder

view live video stream

view live bid stream

place a bid



auctioneer

receive online bid

enter live bids into system

mark item as sold



system

✓ start auction

make payment

✓ track bidder activity

auction  
session

bidder  
tracker



Your Architectural Kata is...

# Going Going Gone!



bidder

view live video stream

view live bid stream

place a bid



video  
streamer



auctioneer

receive online bid

enter live bids into system

mark item as sold



system

✓ start auction

make payment

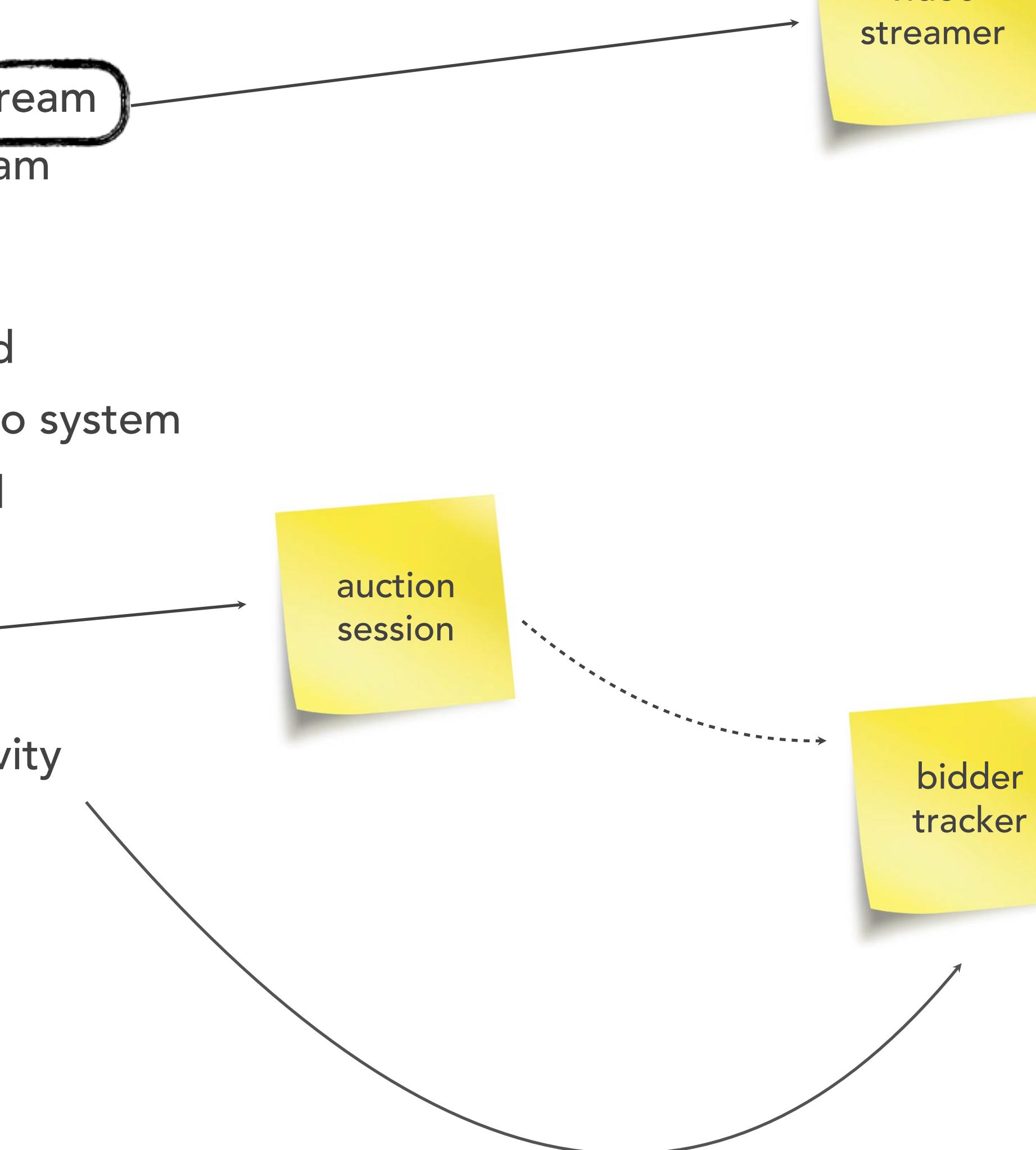
✓ track bidder activity



auction  
session



bidder  
tracker



Your Architectural Kata is...

# Going Going Gone!

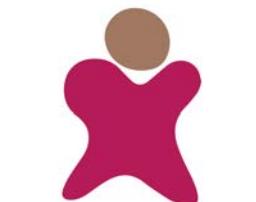


bidder

- ✓ view live video stream
- view live bid stream
- place a bid



video  
streamer



auctioneer

- receive online bid
- enter live bids into system
- mark item as sold



system

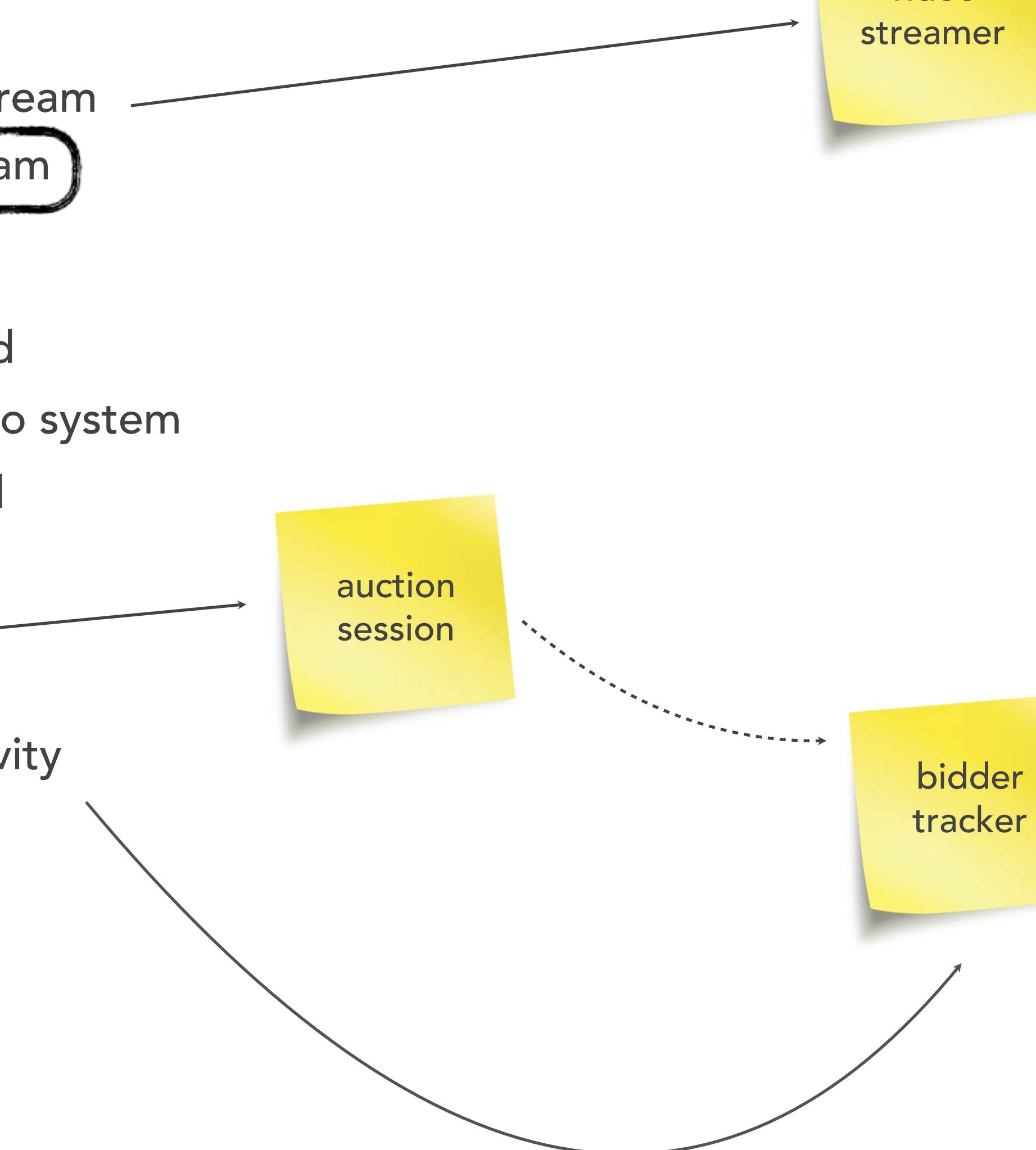
- ✓ start auction
- make payment
- ✓ track bidder activity



auction  
session



bidder  
tracker



Your Architectural Kata is...

# Going Going Gone!



bidder

- ✓ view live video stream
- view live bid stream
- place a bid



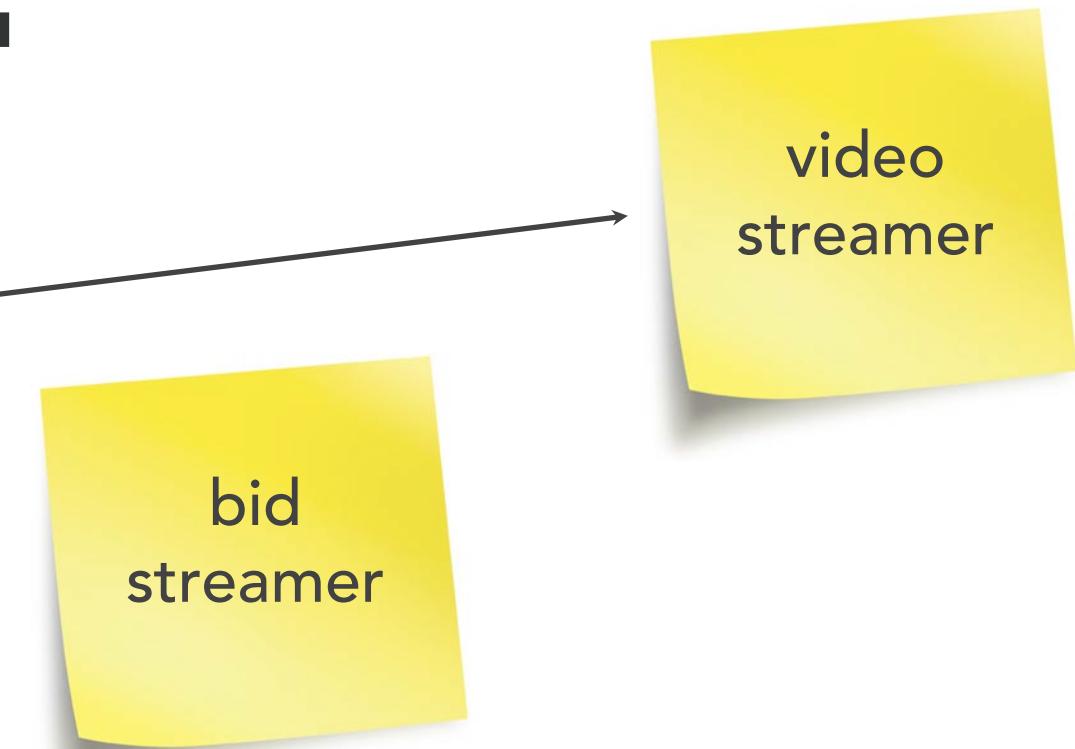
auctioneer

- receive online bid
- enter live bids into system
- mark item as sold



system

- ✓ start auction
- make payment
- ✓ track bidder activity



Your Architectural Kata is...

# Going Going Gone!



bidder

- ✓ view live video stream
- ✓ view live bid stream
- place a bid



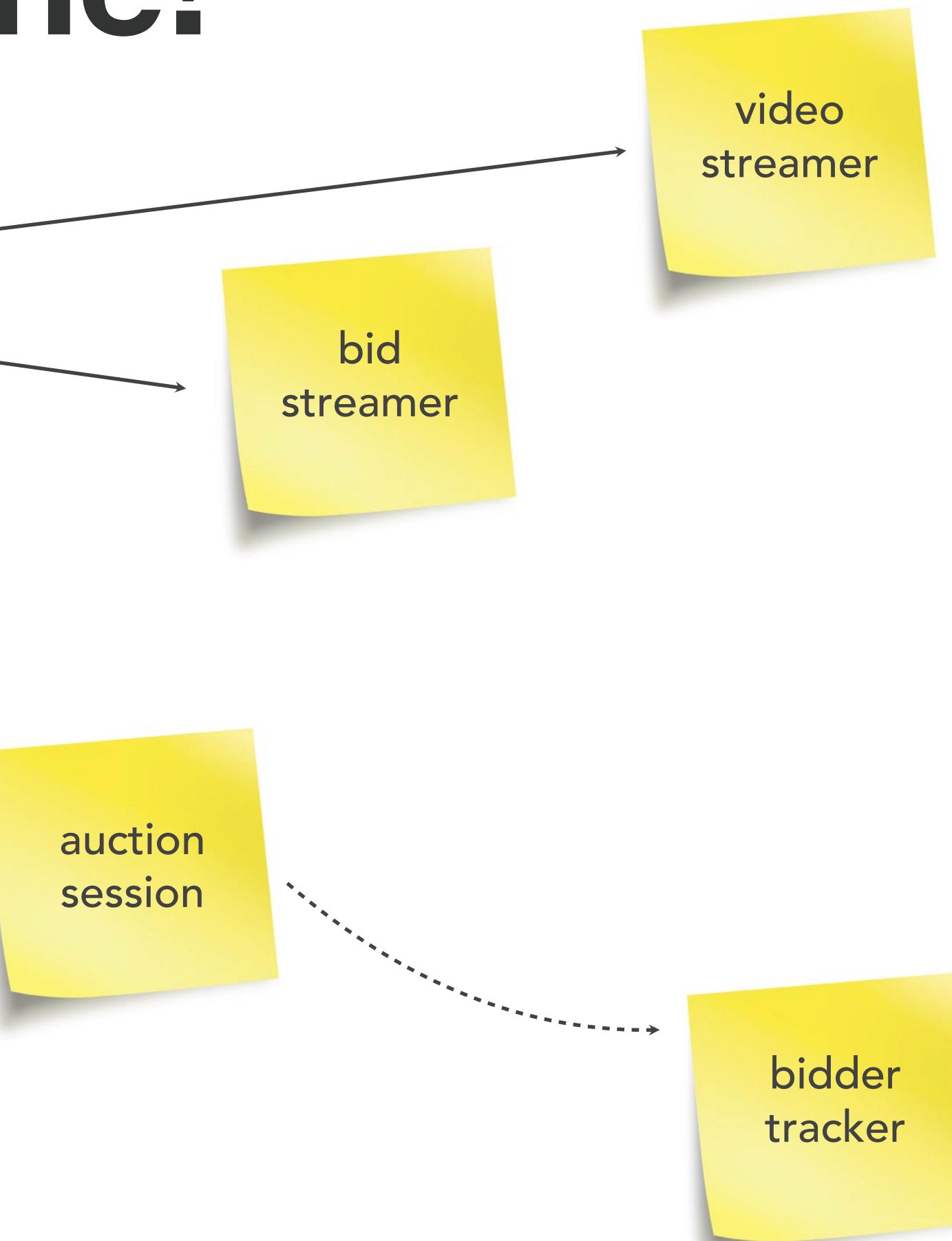
auctioneer

- receive online bid
- enter live bids into system
- mark item as sold



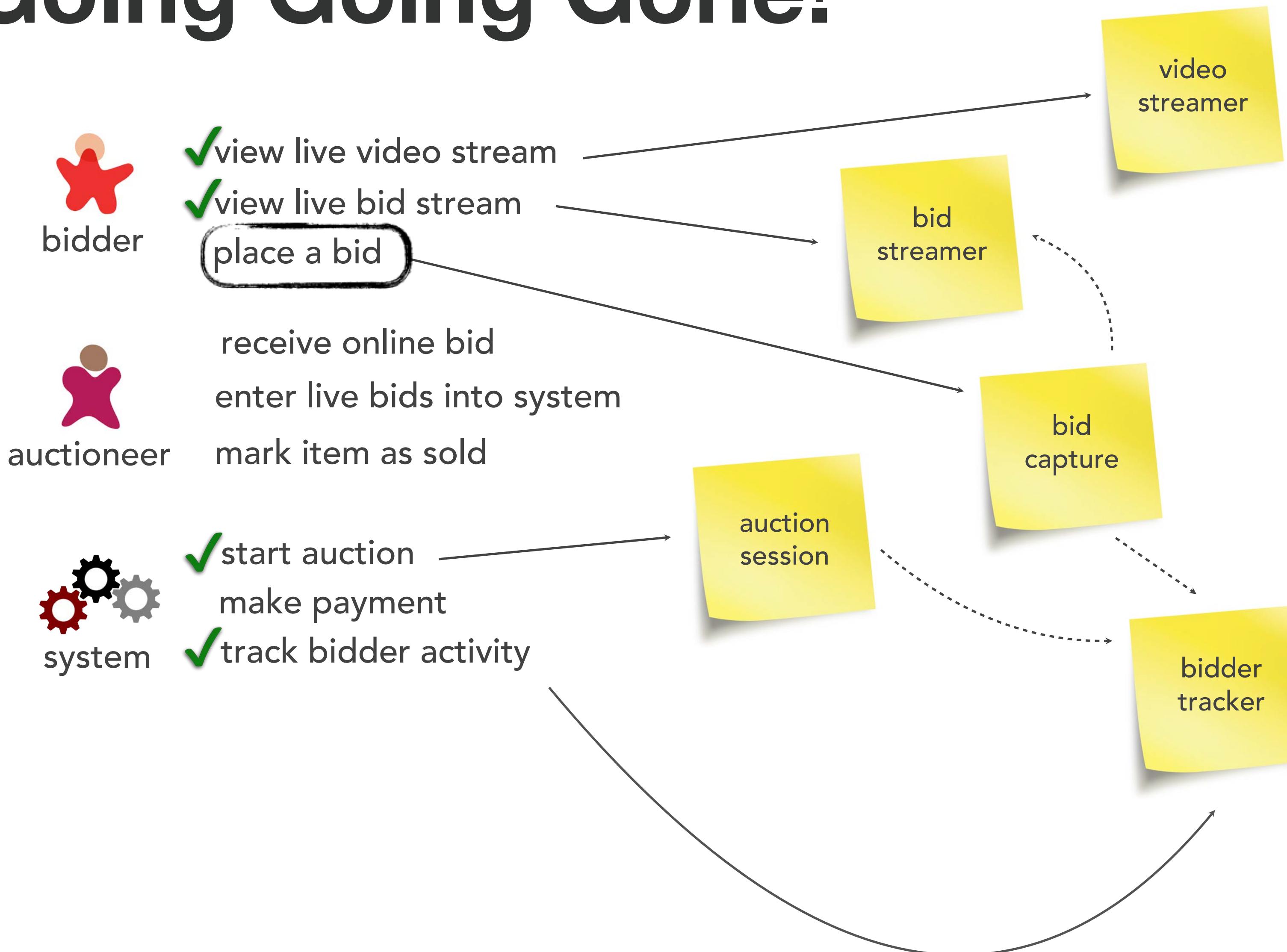
system

- ✓ start auction
- make payment
- ✓ track bidder activity



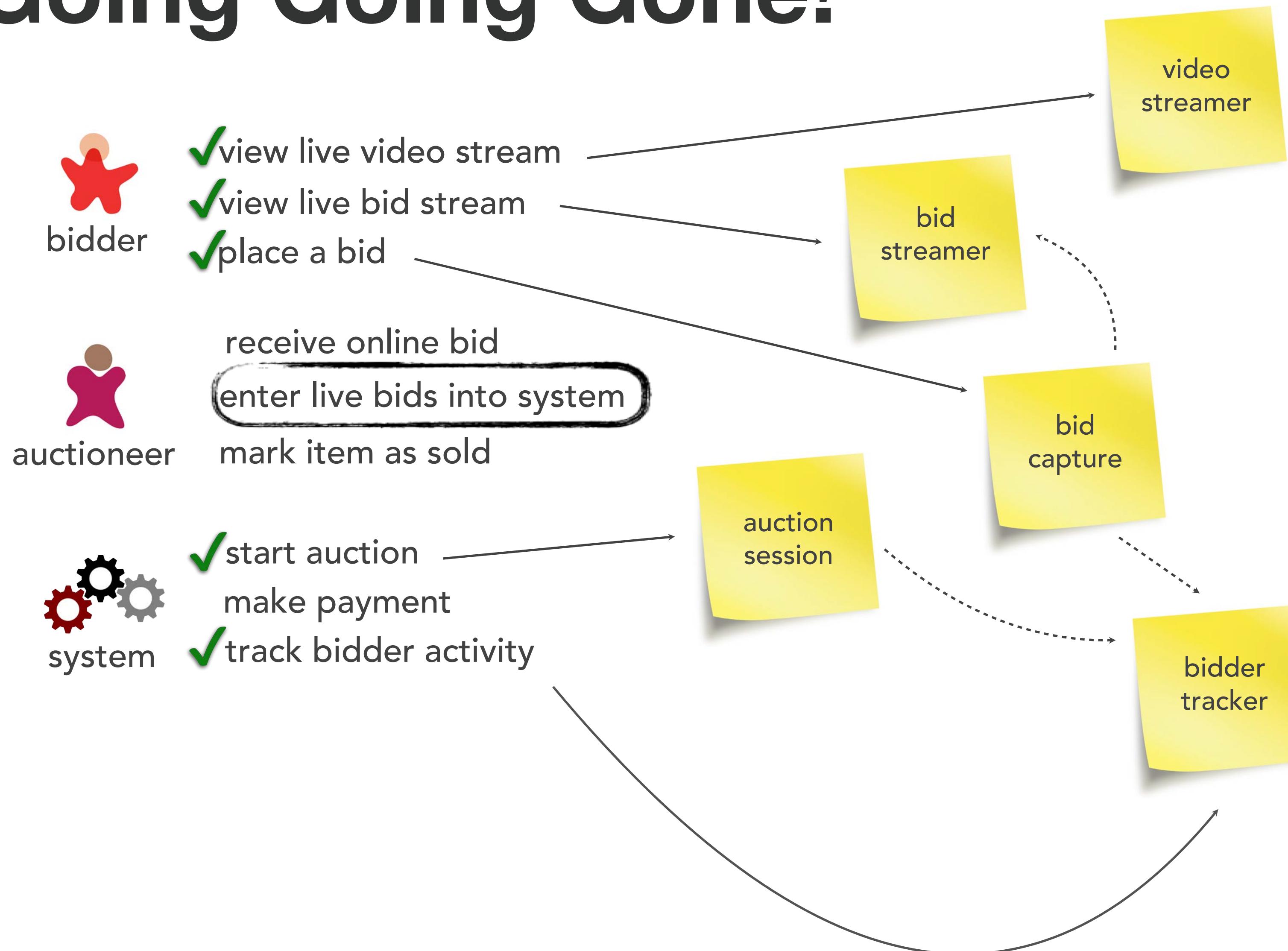
Your Architectural Kata is...

# Going Going Gone!



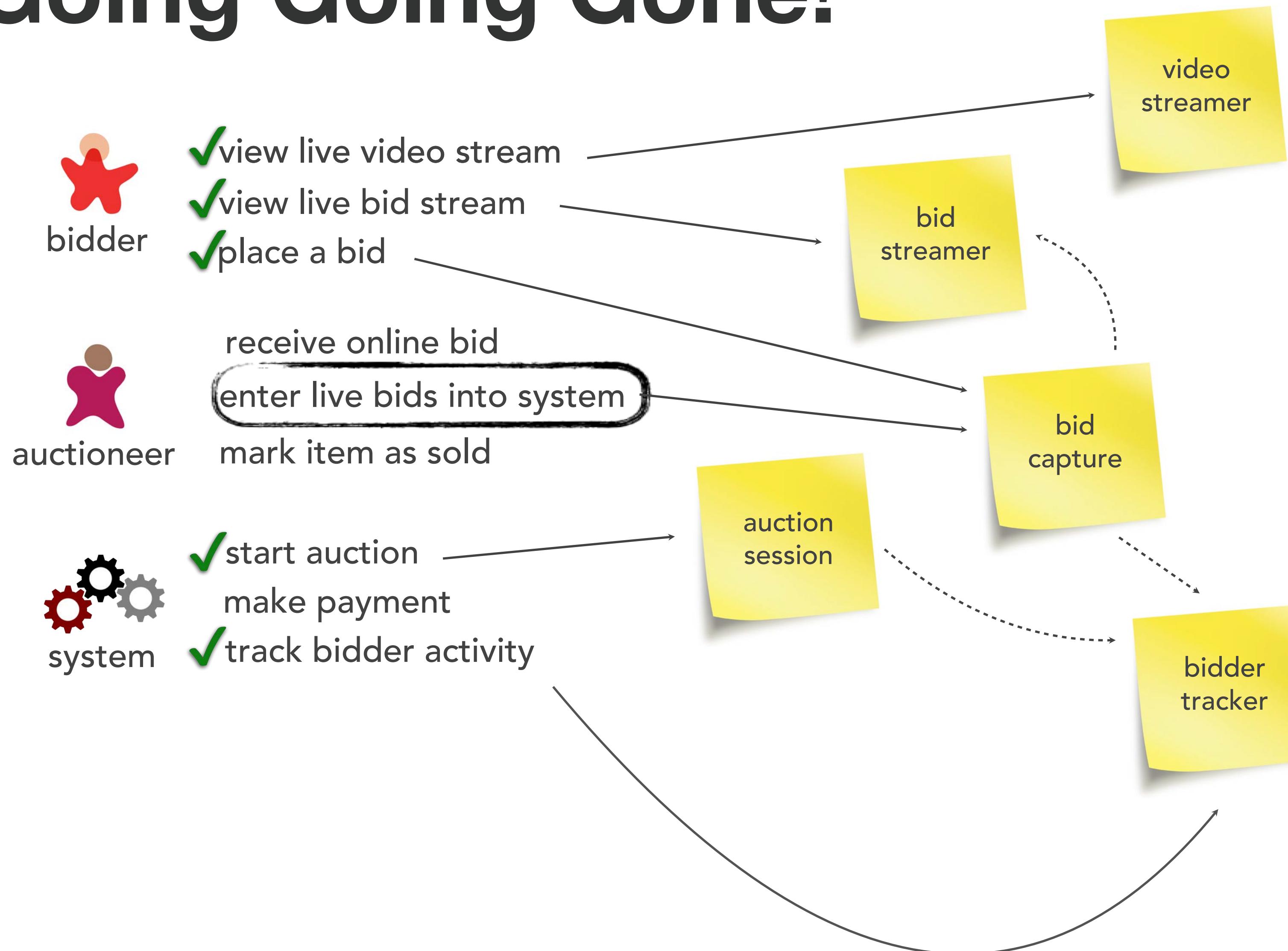
Your Architectural Kata is...

# Going Going Gone!



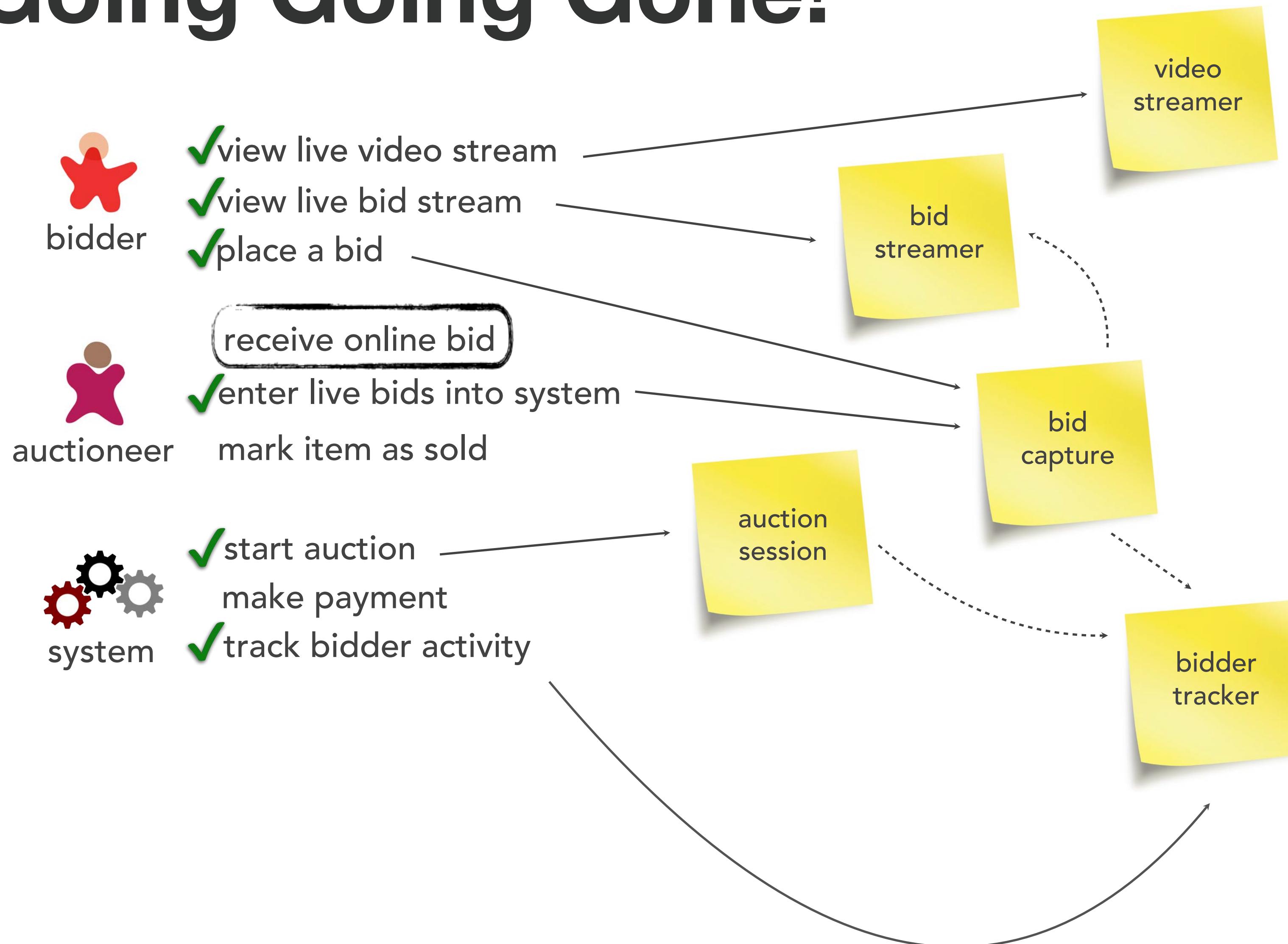
Your Architectural Kata is...

# Going Going Gone!



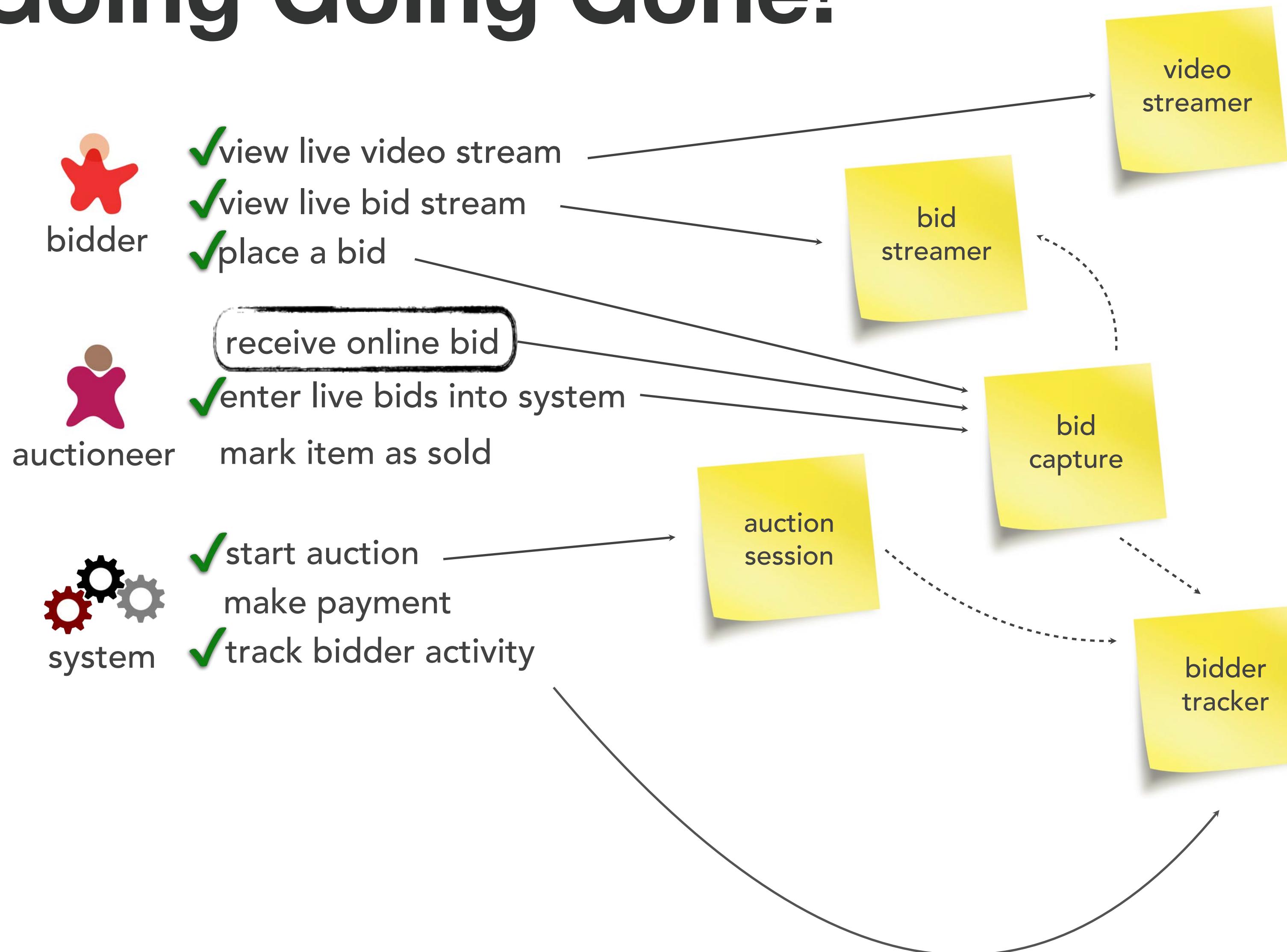
Your Architectural Kata is...

# Going Going Gone!



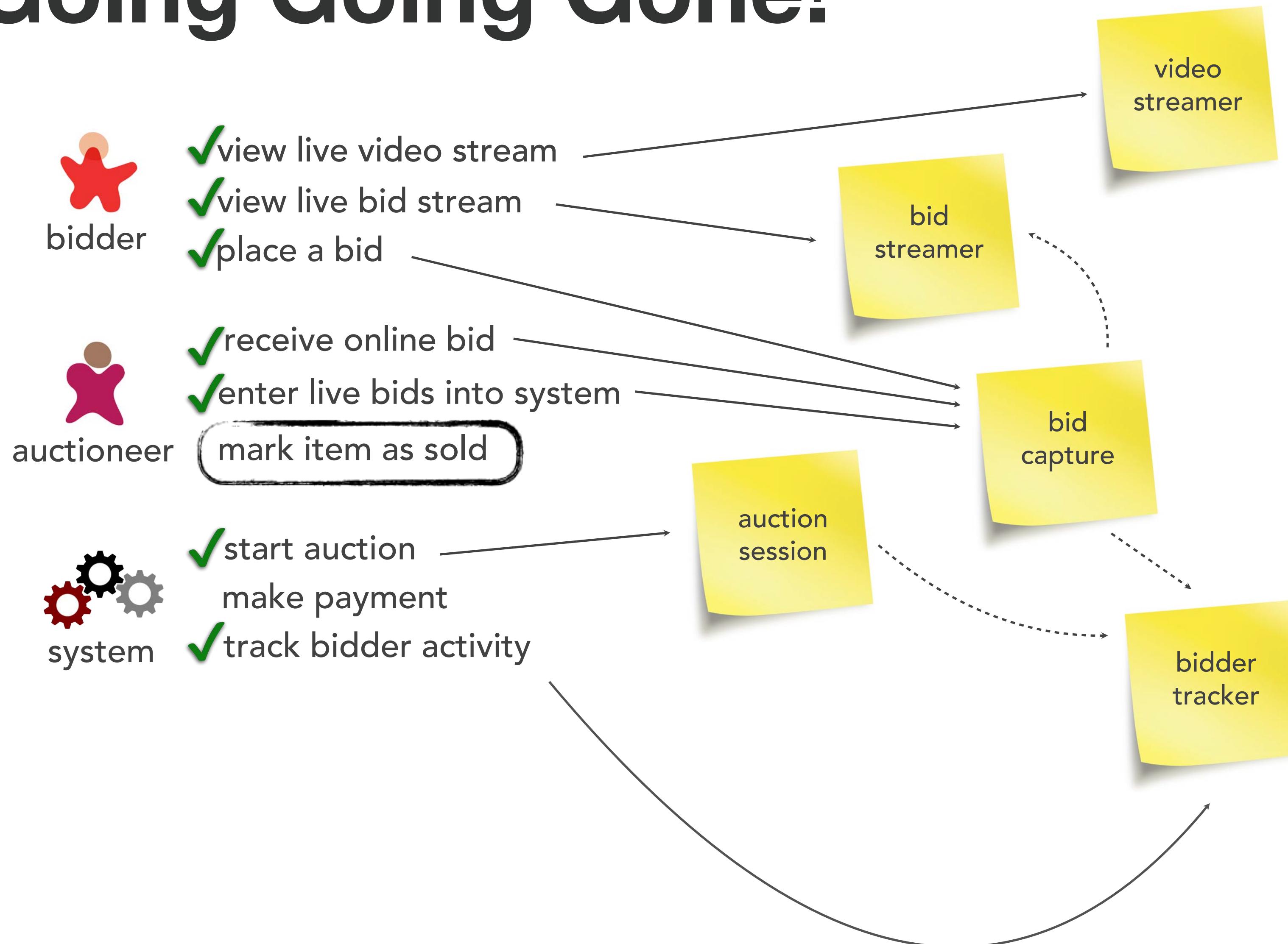
Your Architectural Kata is...

# Going Going Gone!



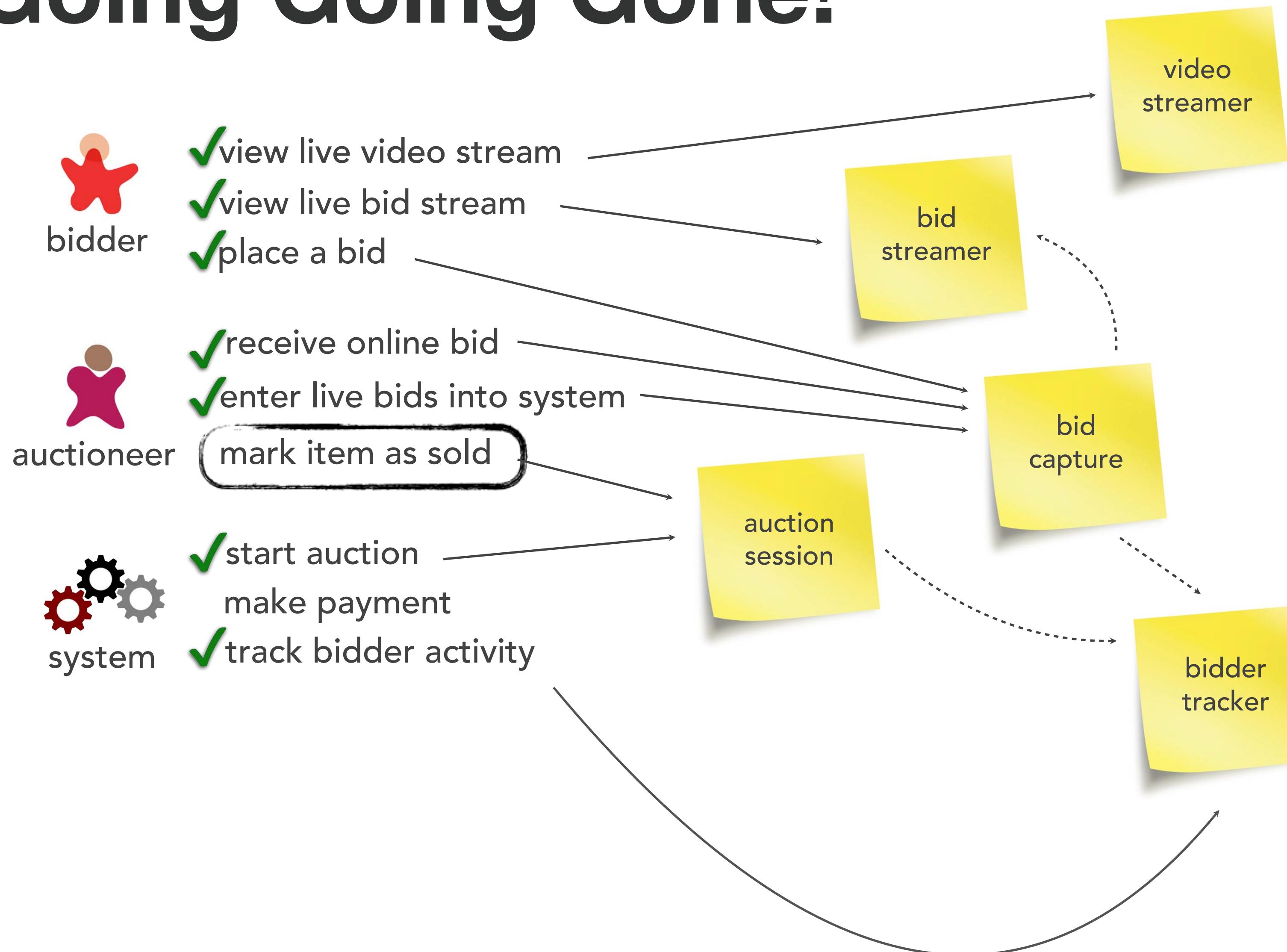
Your Architectural Kata is...

# Going Going Gone!



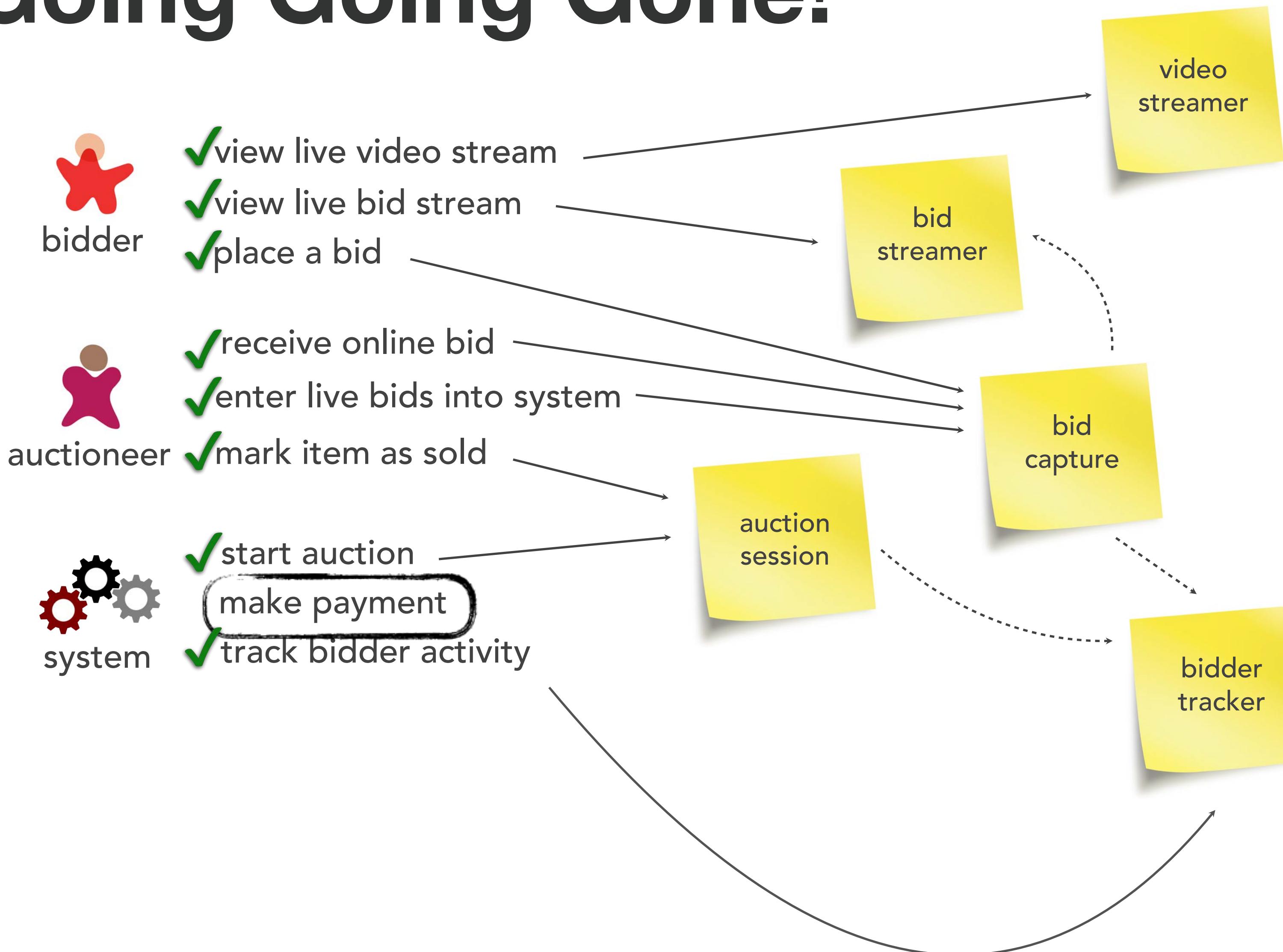
Your Architectural Kata is...

# Going Going Gone!



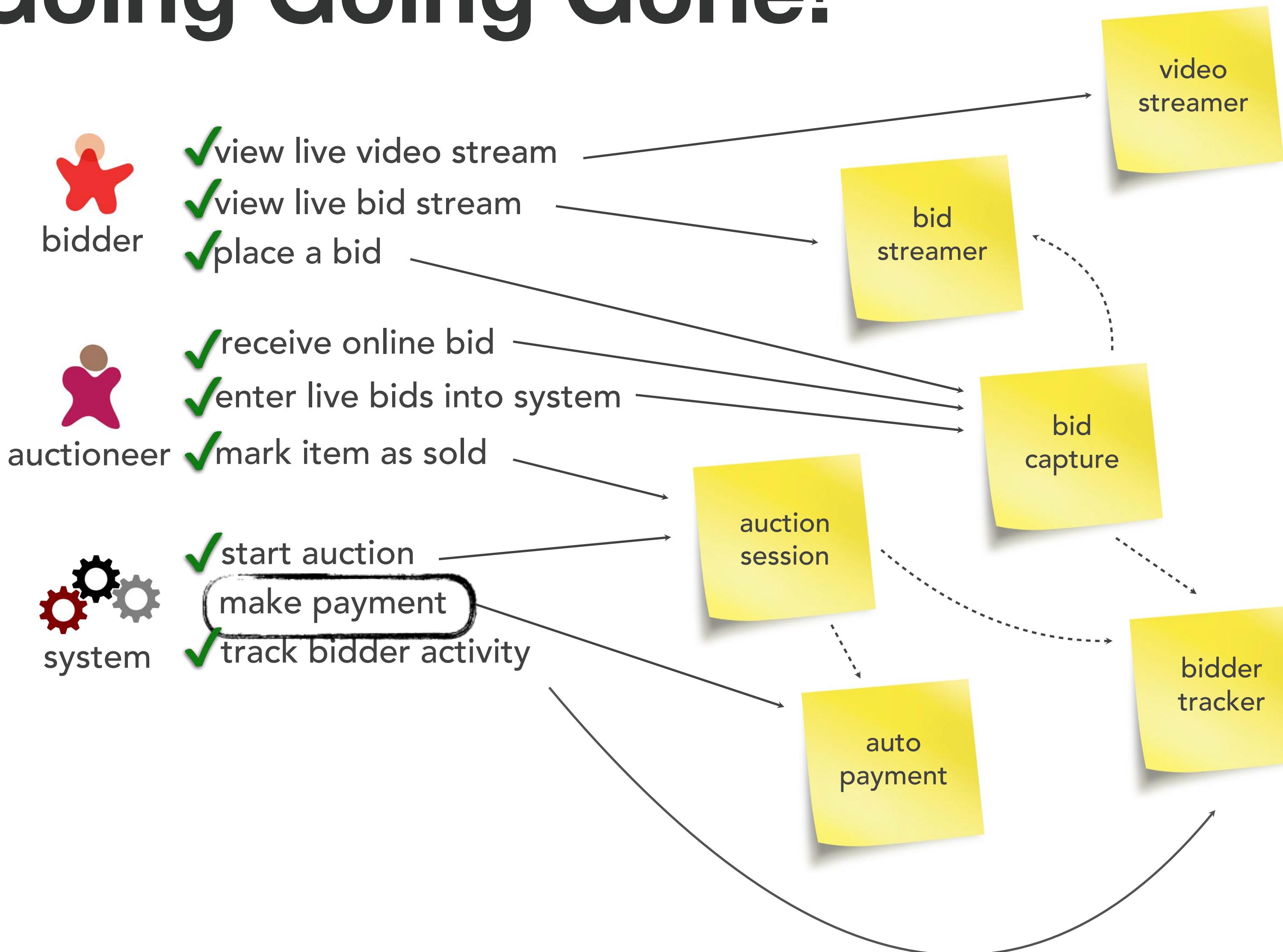
Your Architectural Kata is...

# Going Going Gone!



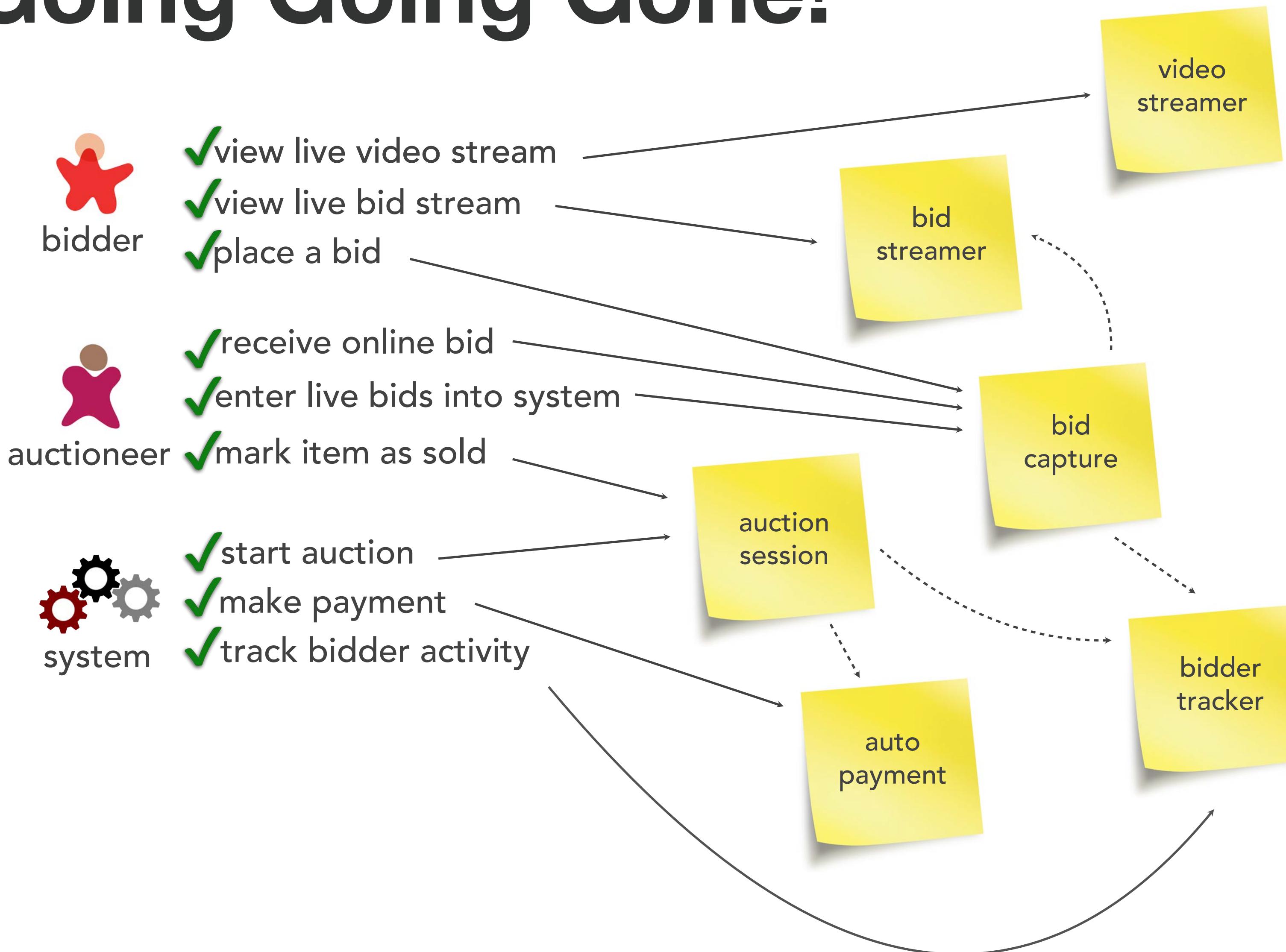
Your Architectural Kata is...

# Going Going Gone!



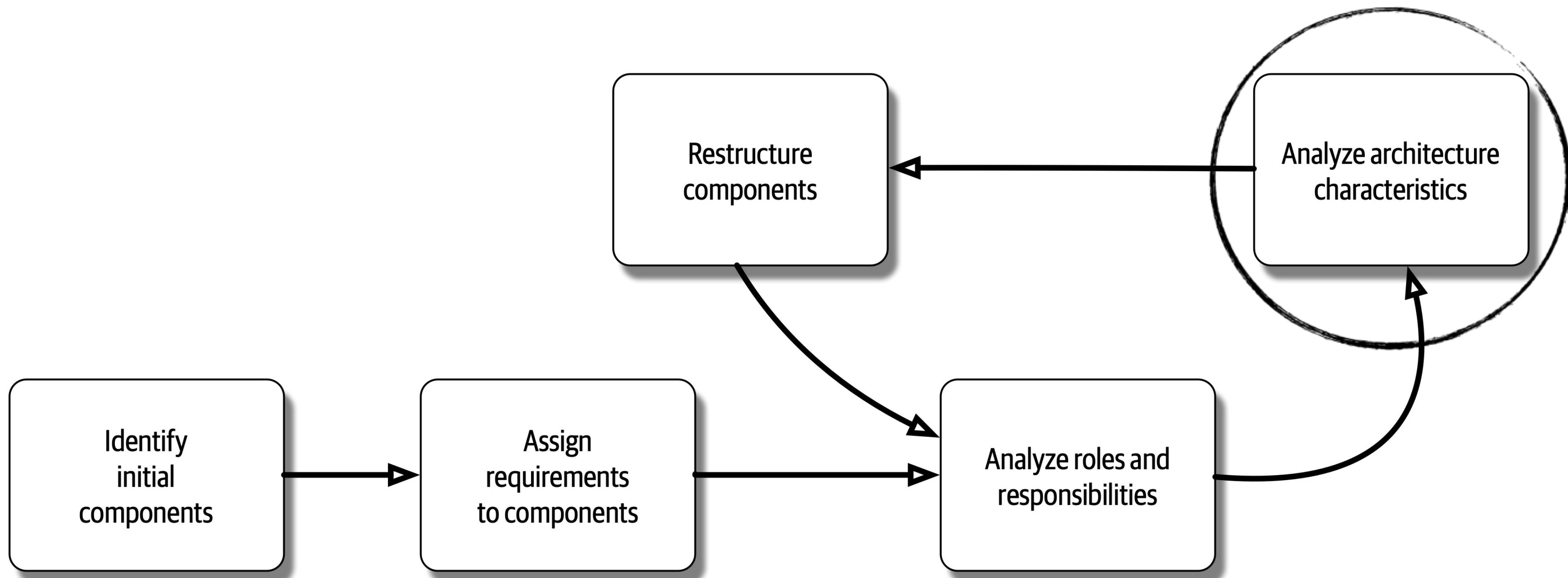
Your Architectural Kata is...

# Going Going Gone!



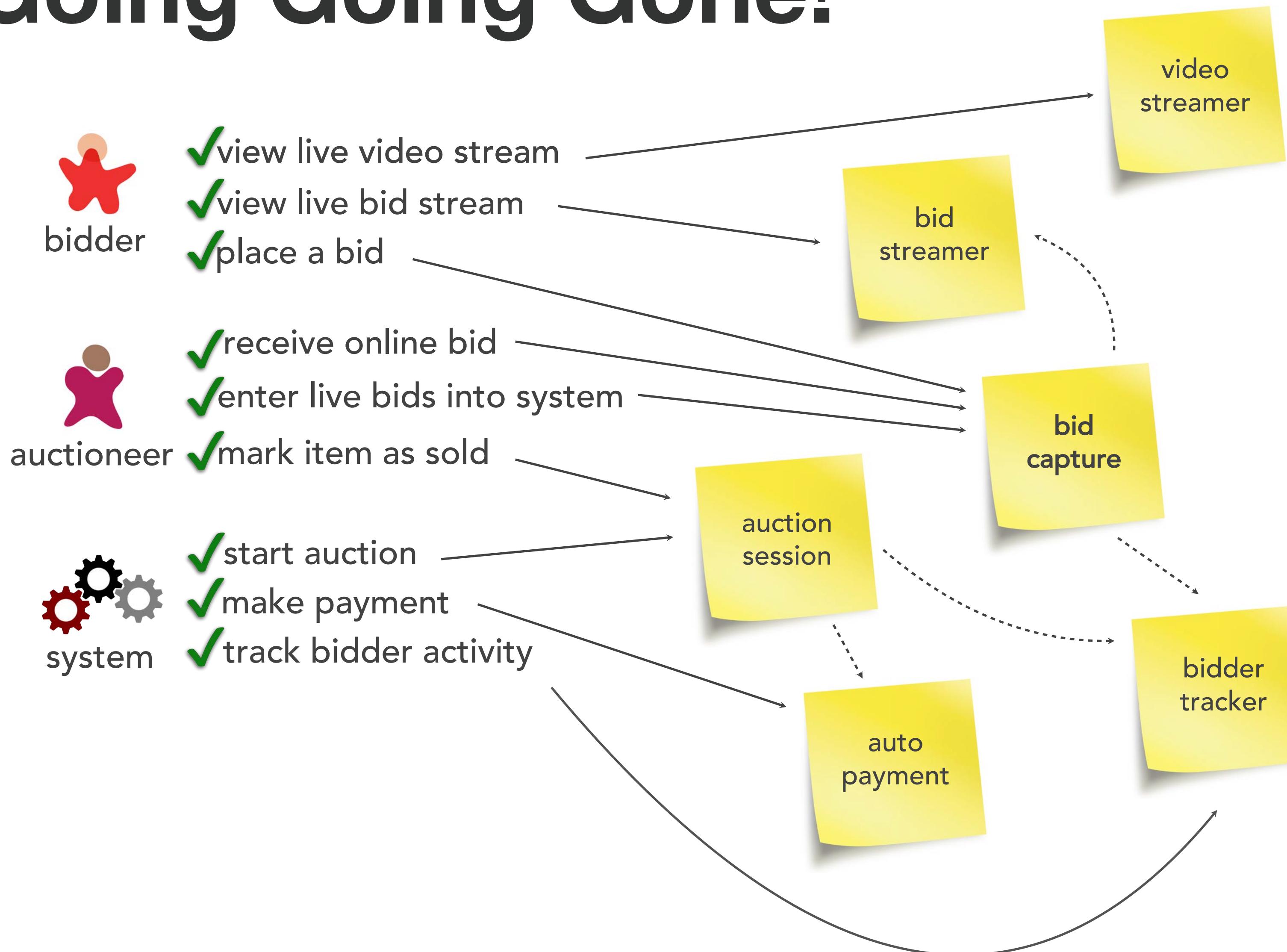
Your Architectural Kata is...

# Going Going Gone!



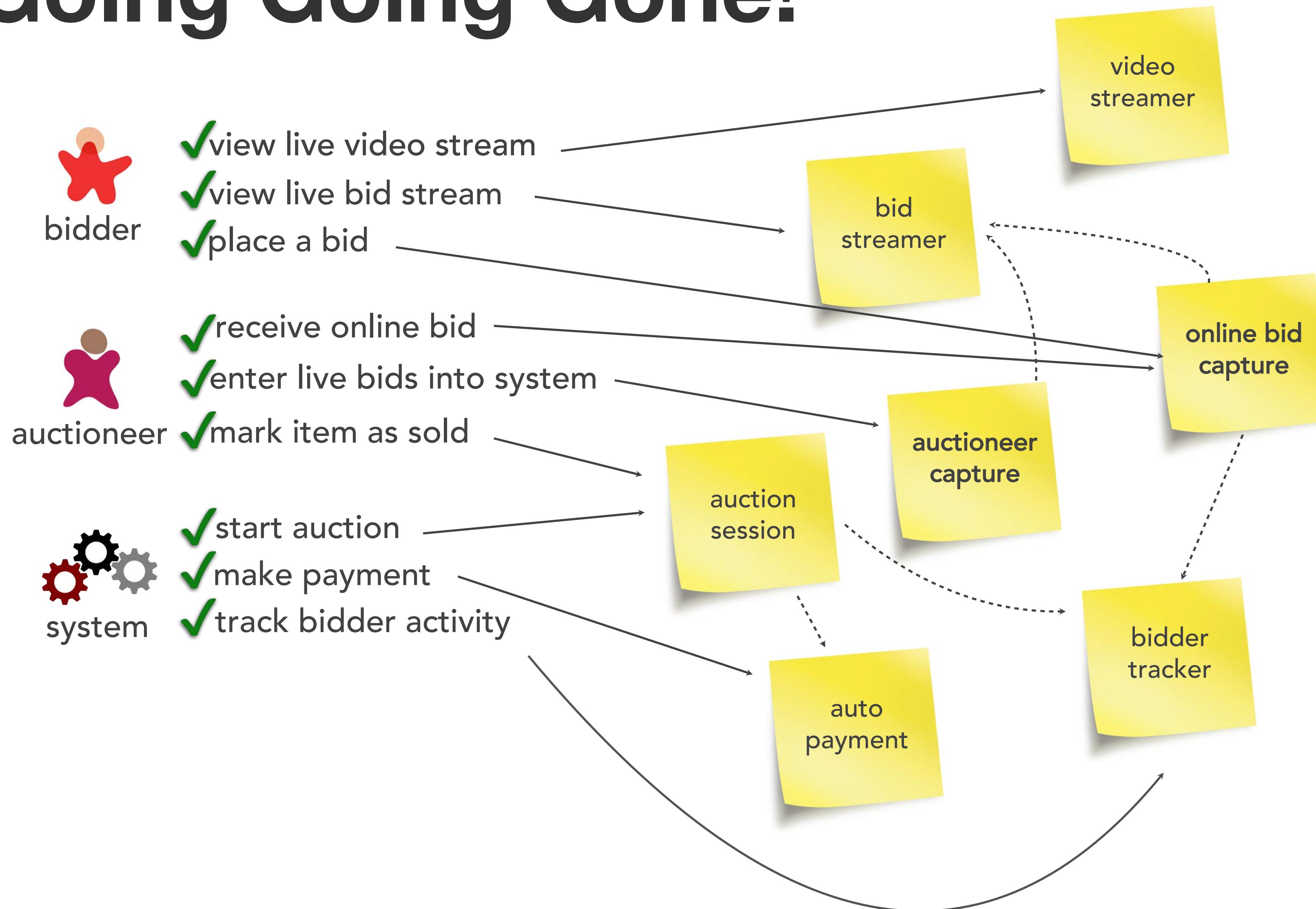
Your Architectural Kata is...

# Going Going Gone!



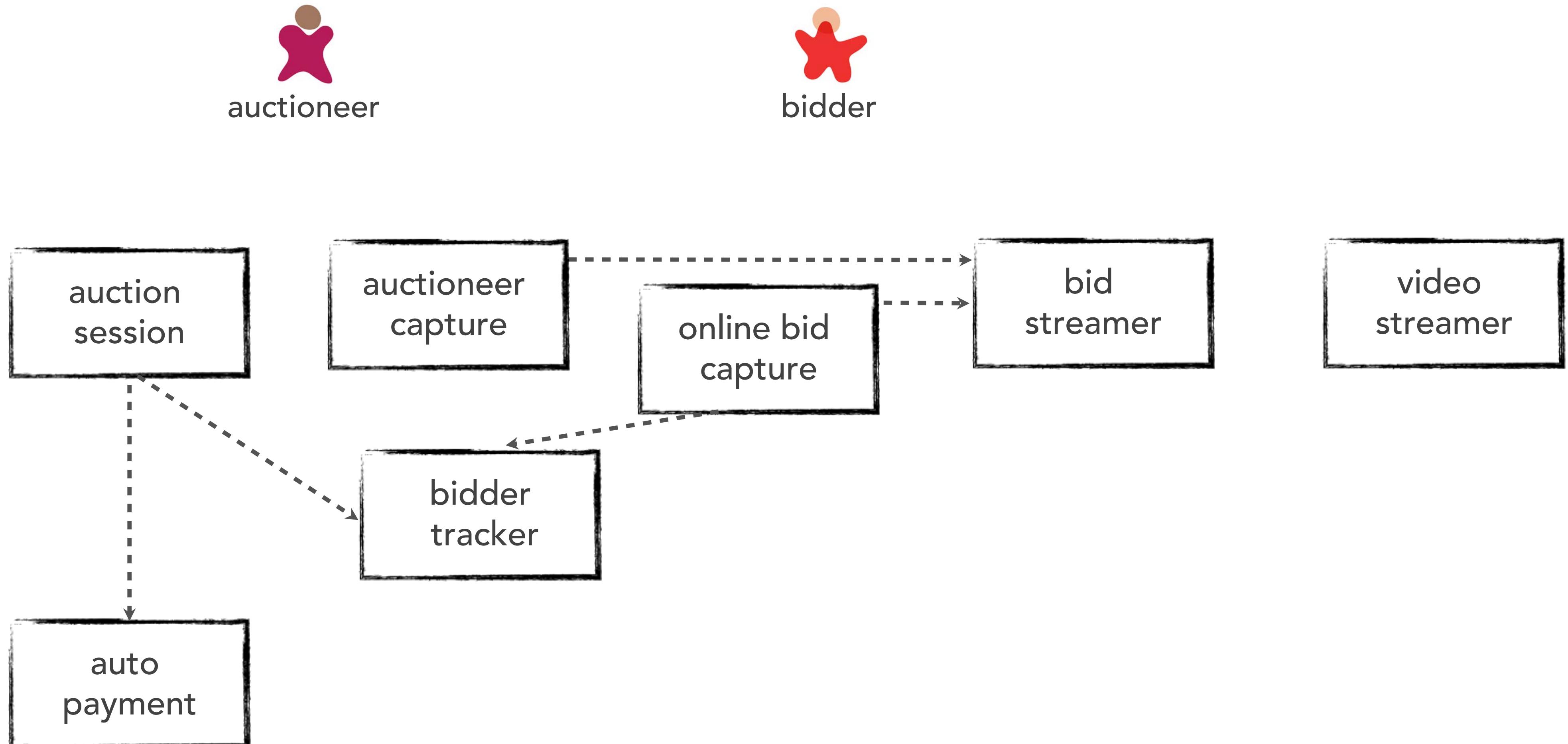
Your Architectural Kata is...

# Going Going Gone!



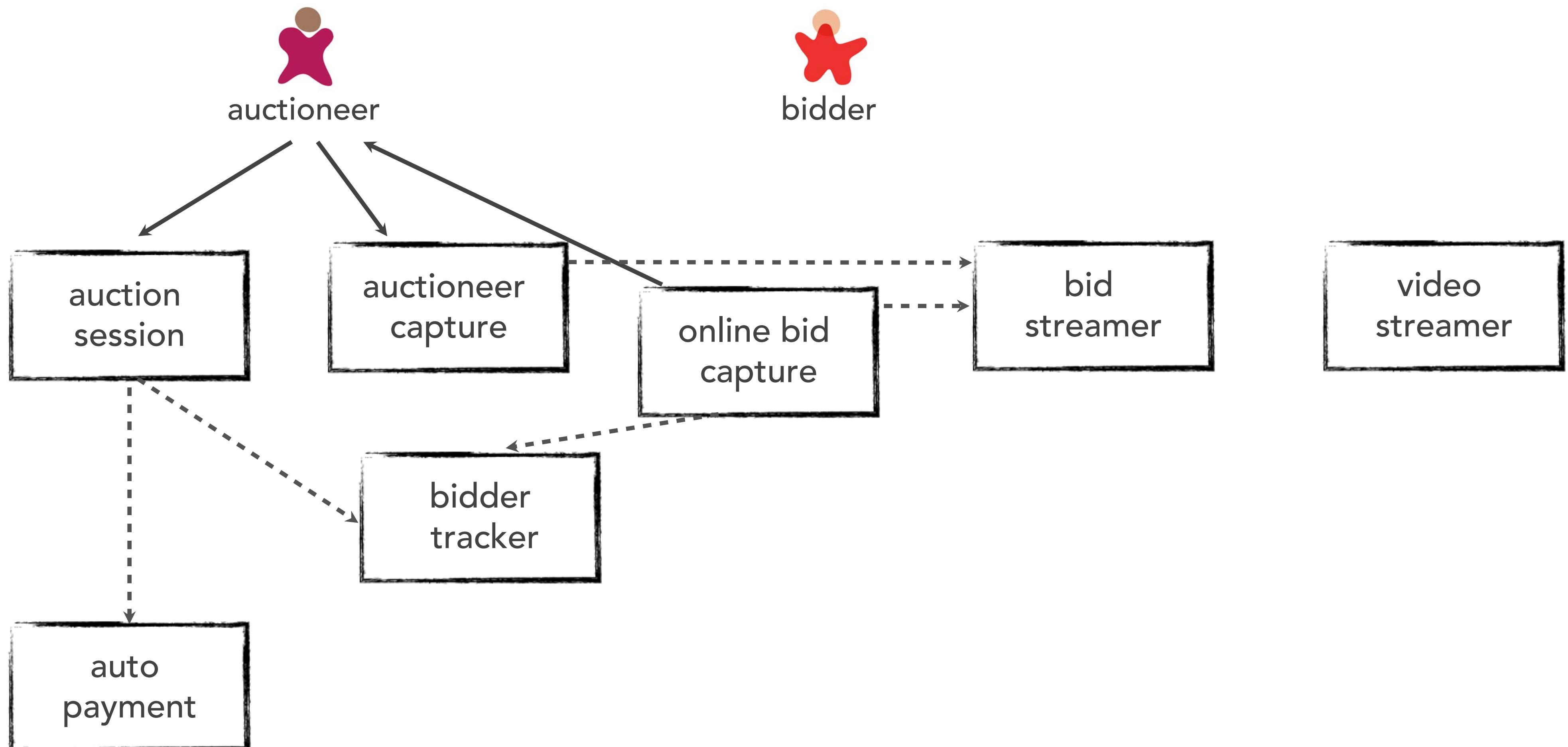
Your Architectural Kata is...

# Going Going Gone!



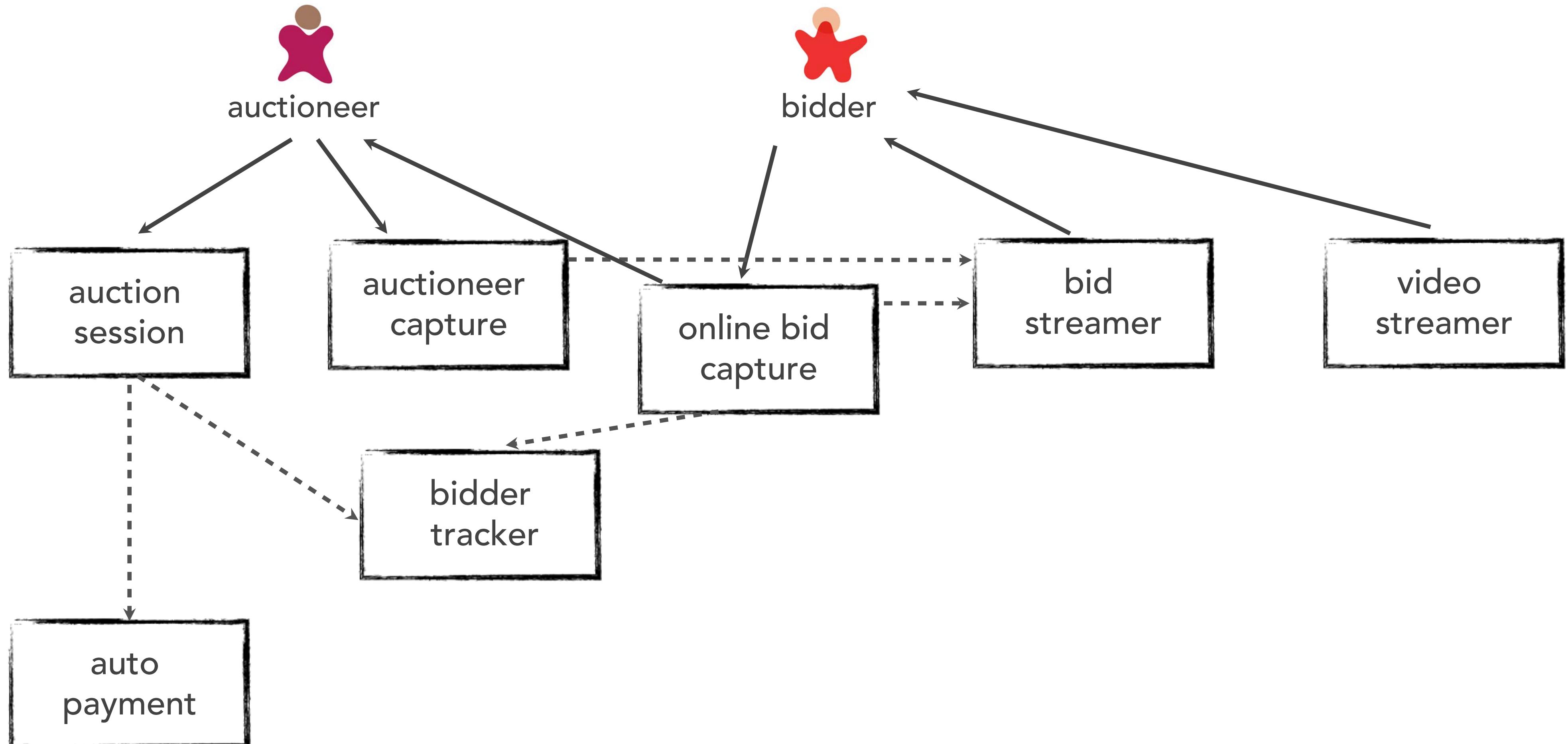
Your Architectural Kata is...

# Going Going Gone!



Your Architectural Kata is...

# Going Going Gone!



Your Architectural Kata is...

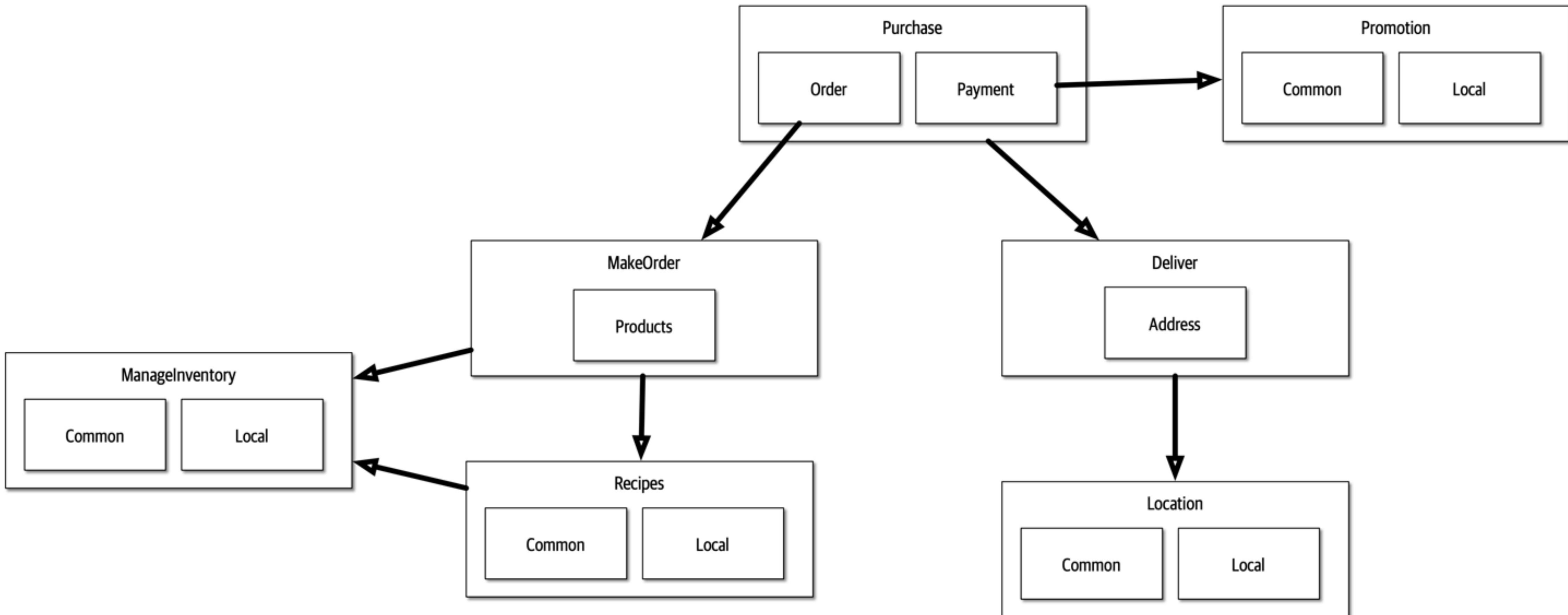
# Silicon Sandwiches

A national sandwich shop wants to enable internet-ordering (in addition to their current call-in service)

- ***Users:*** thousands, perhaps one day millions
- ***Requirements:***
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotions/specials
  - offer local daily promotions/specials
  - accept payment online or in person/on delivery
- ***Additional Context:***
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.
  - Time to market is critical.

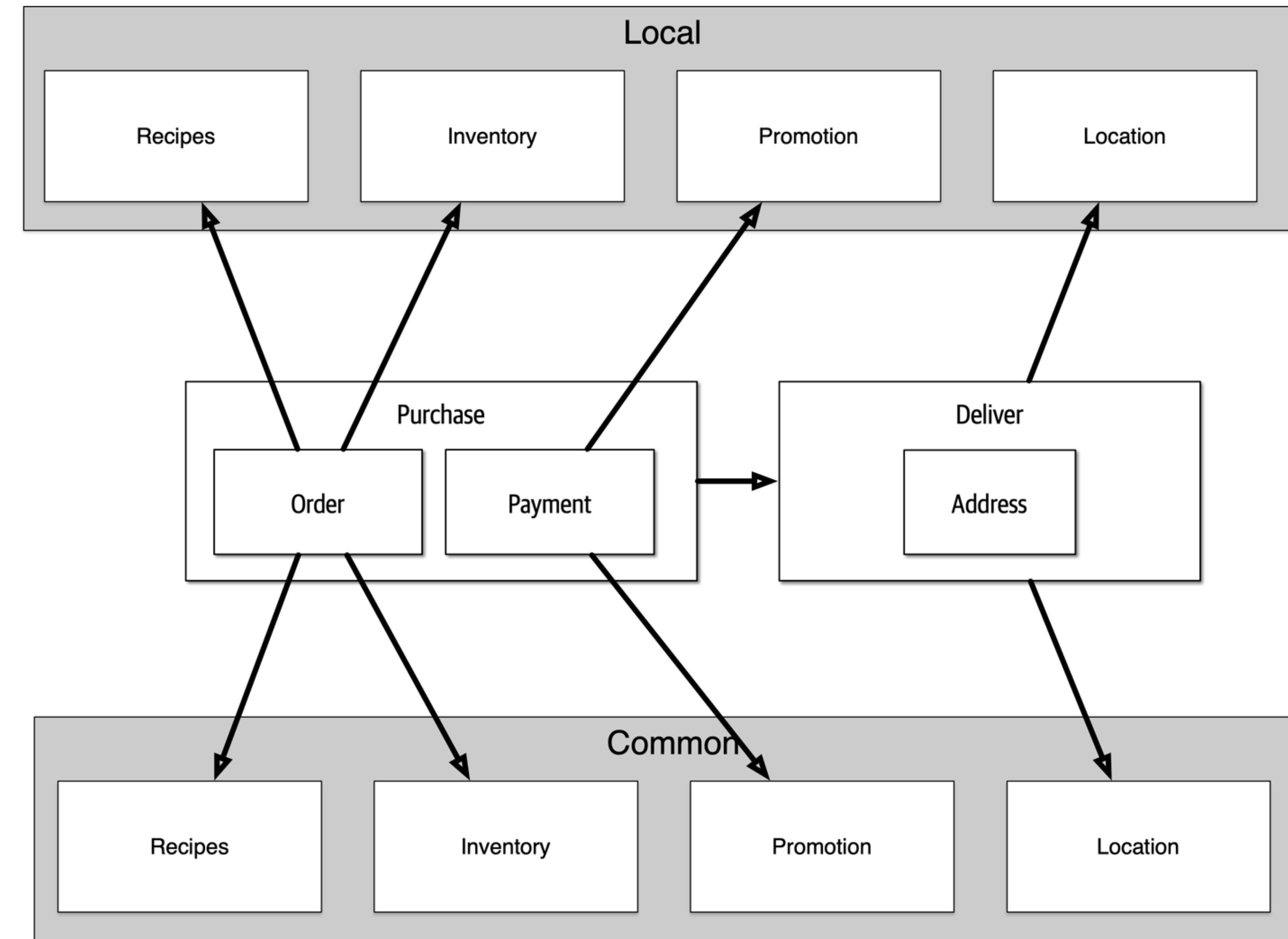
Your Architectural Kata is...

# Silicon Sandwiches



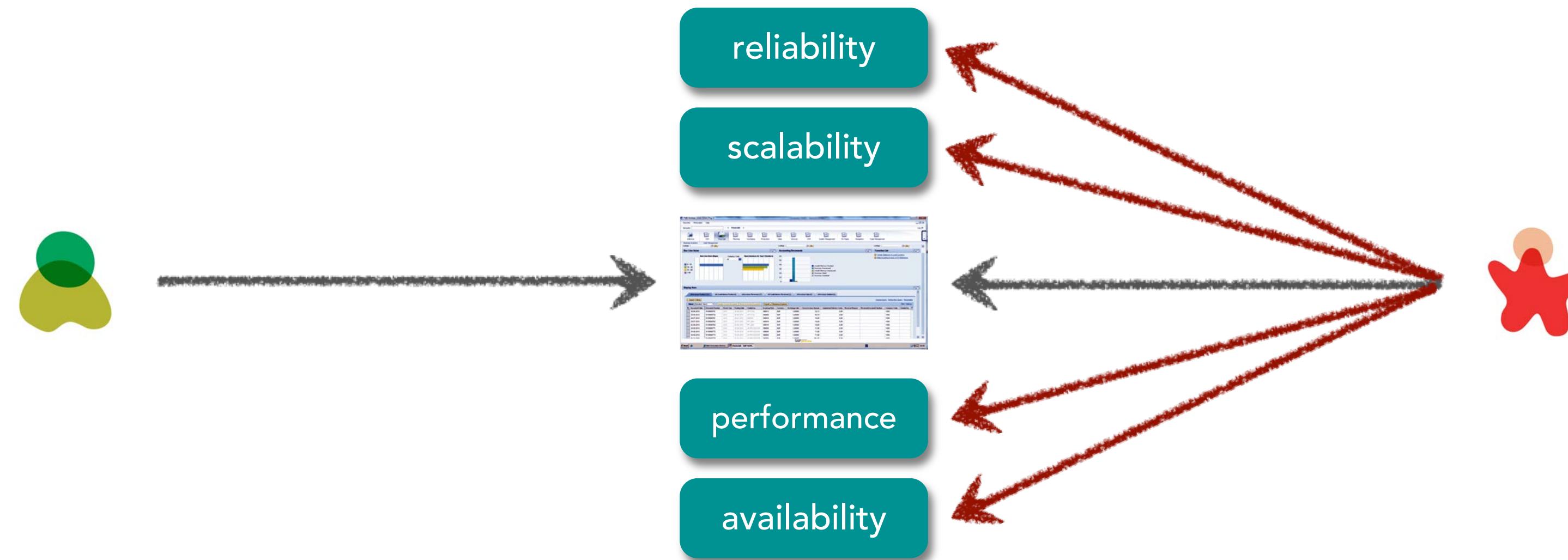
Your Architectural Kata is...

# Silicon Sandwiches



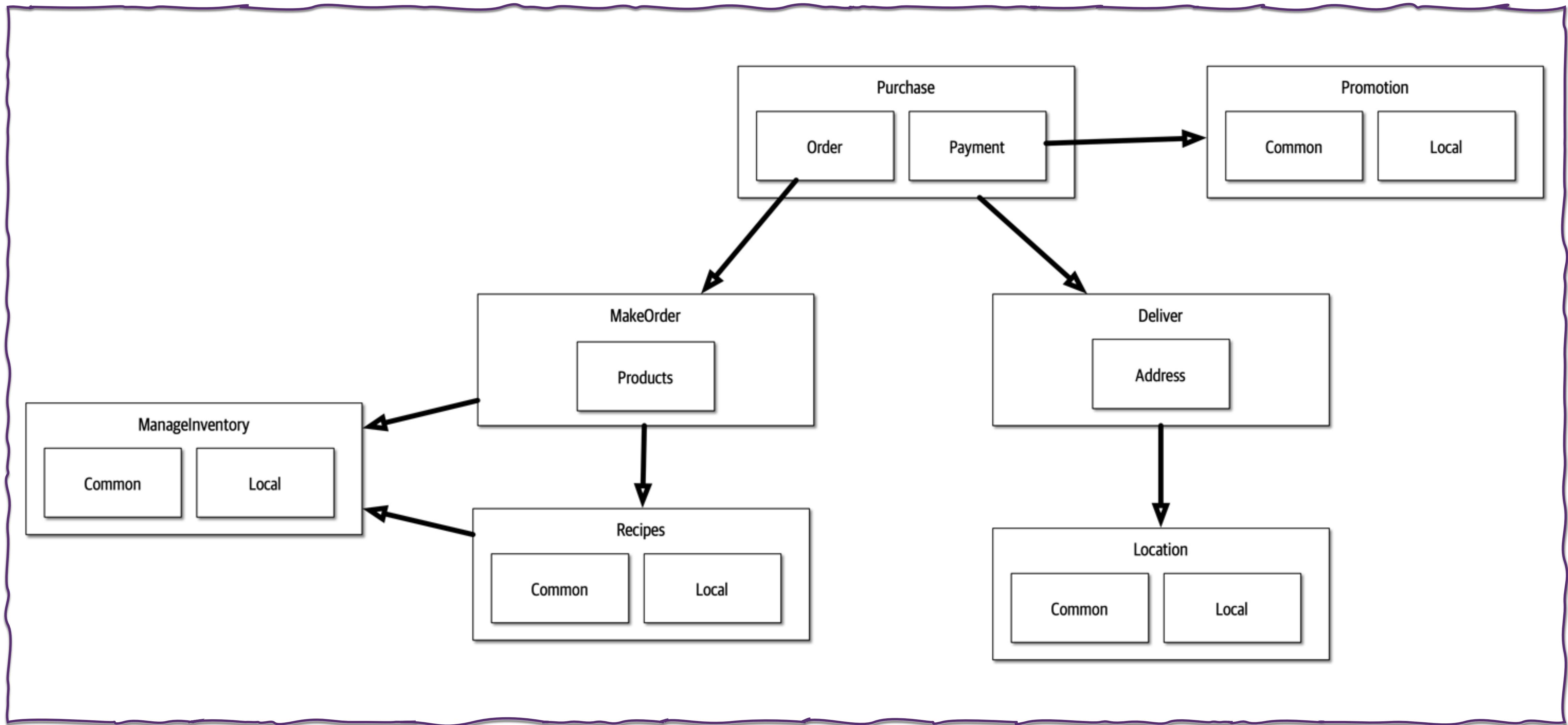
# architecture characteristics scope?

Differing architecture characteristics  
for groups of components =>  
distributed architecture.



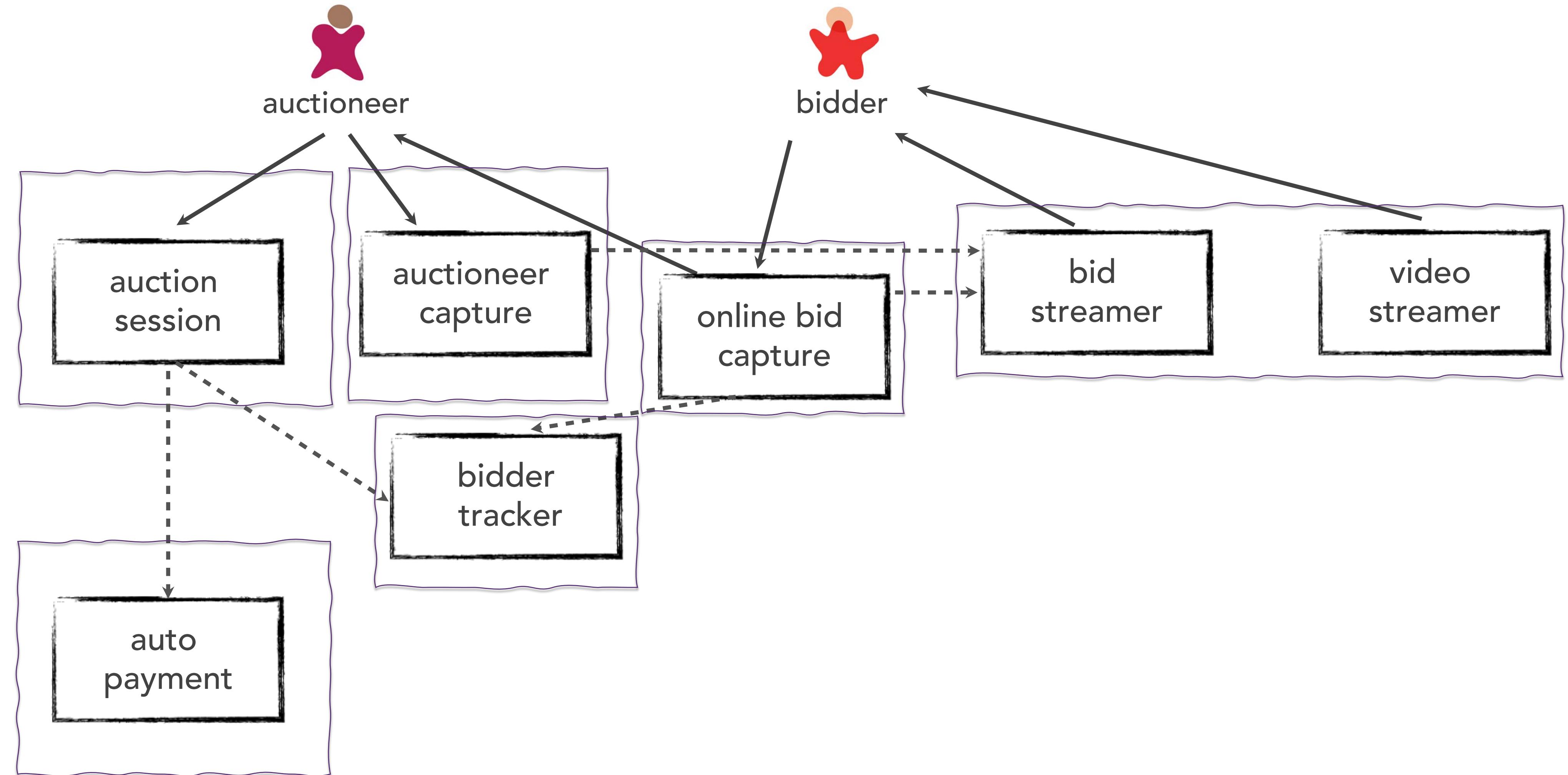
Your Architectural Kata is...

# Silicon Sandwiches



Your Architectural Kata is...

# Going Going Gone!



# architecture katas

## identifying architecture styles

Your Architectural Kata is...

### Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - video stream of the action after the fact
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

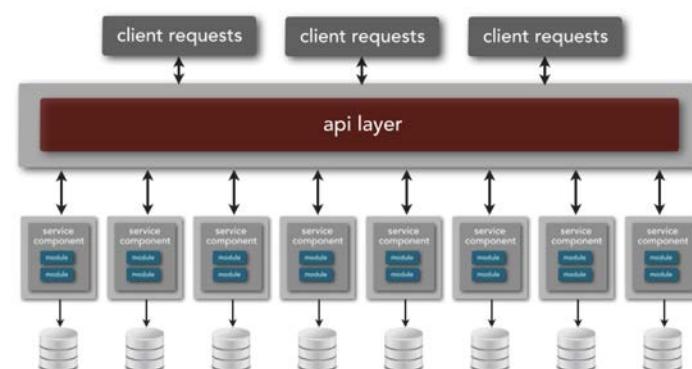
Your Architectural Kata is...

### Silicon Sandwiches

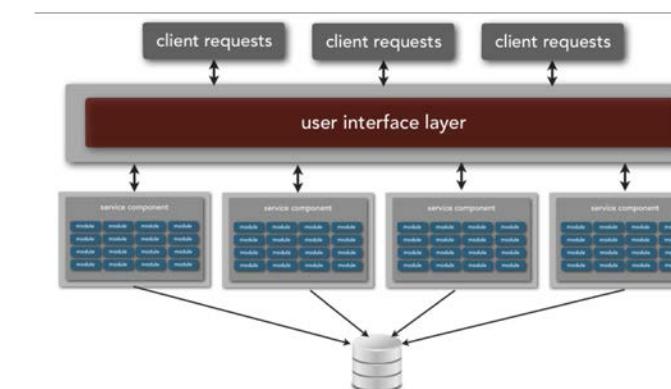
A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- **Users:** thousands, perhaps one day millions
- **Requirements:**
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotional/specials
  - offer local daily promotional/specials
  - accept payment online or in person/on delivery
- **Additional Context:**
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.

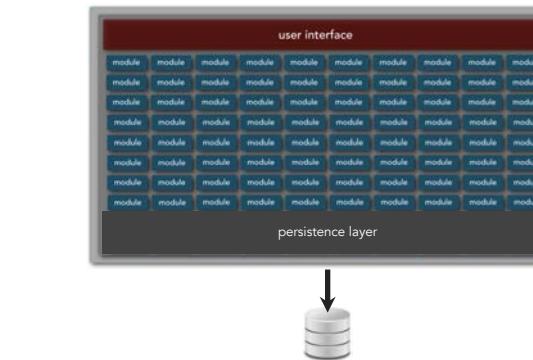
# architecture styles help define the basic characteristics and behavior of an application



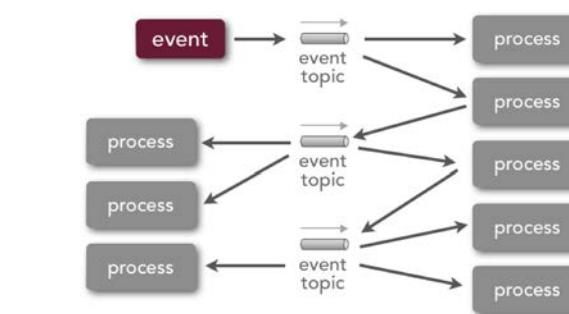
microservices  
architecture



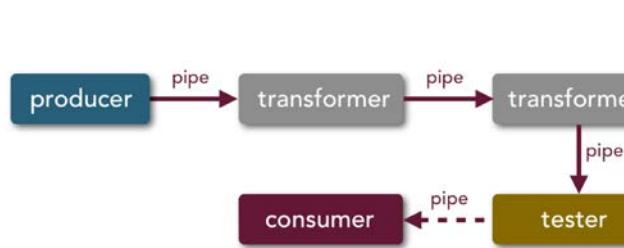
service-based  
architecture



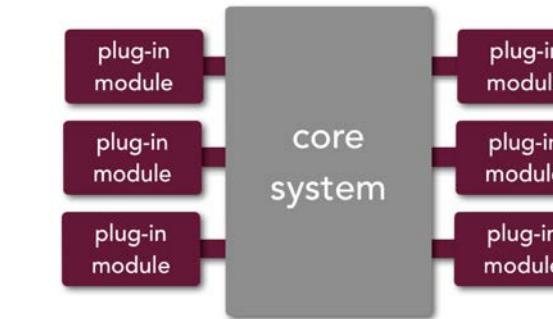
layered  
architecture



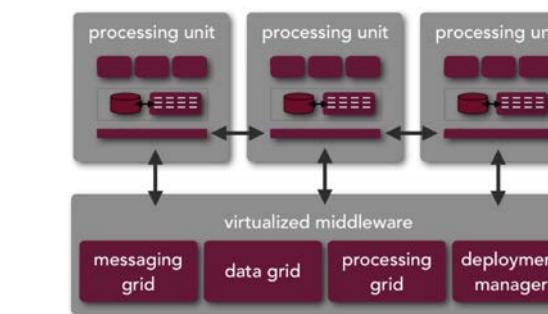
event-driven  
architecture



pipeline  
architecture



microkernel  
architecture



space-based  
architecture

Your Architectural Kata is...

# Going Going Gone!

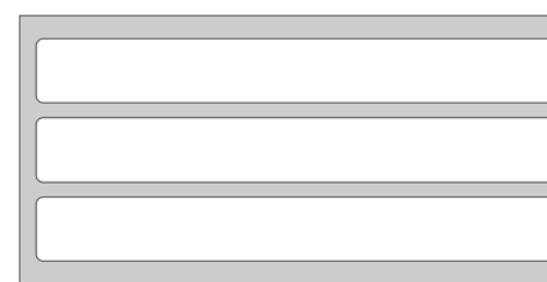
An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

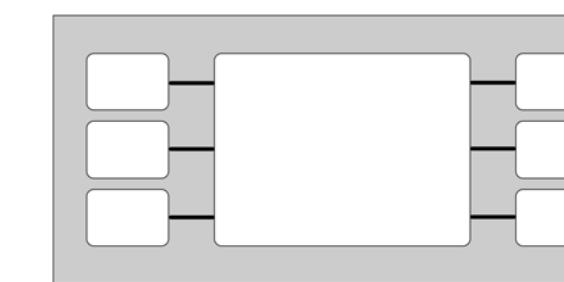
concurrency

availability reliability performance scalability elasticity (security)

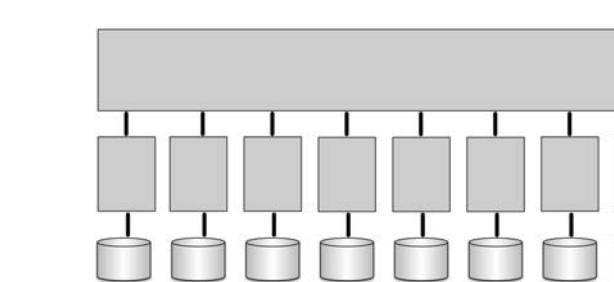
# Going Going Gone!



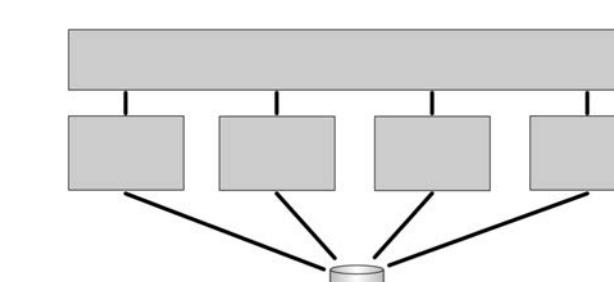
layered monolith



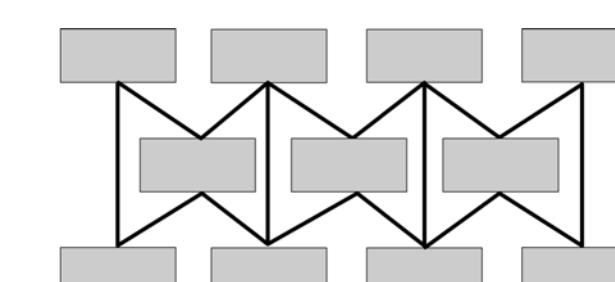
microkernel



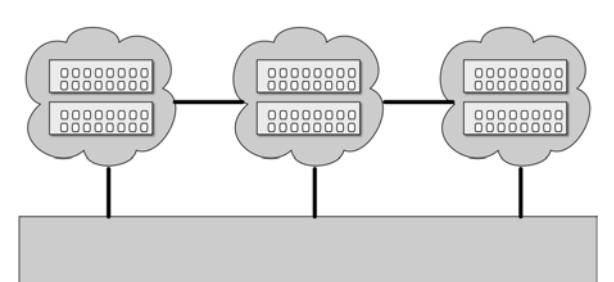
microservices



service-based



event-driven



space-based

	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★★	★★★★★	★★★★★	★★★★	★★★★
testability	★★	★★★★	★★★★★	★★★★	★★★	★
performance	★★★★★	★★★★★	★	★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★★★	★★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★
evolvability	★	★★★★	★★★★★	★★★★★	★★★★★	★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★

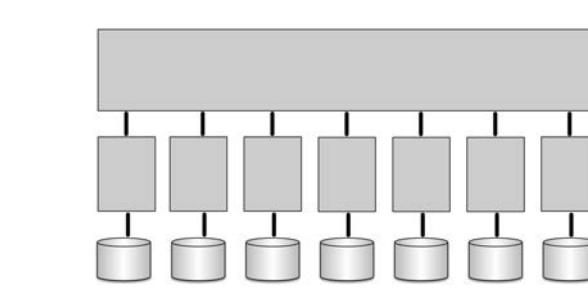
# Going Going Gone!



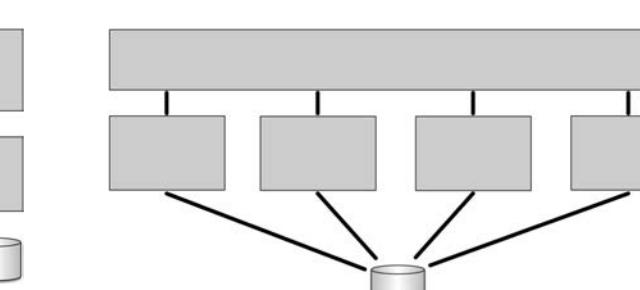
layered monolith



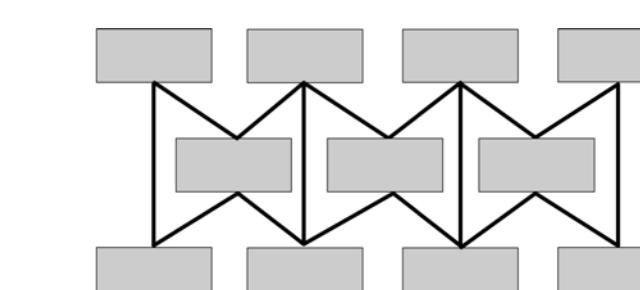
microkernel



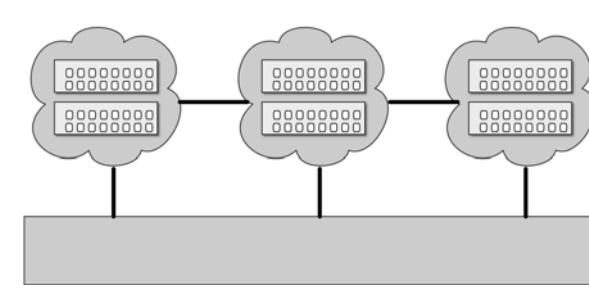
microservices



service-based



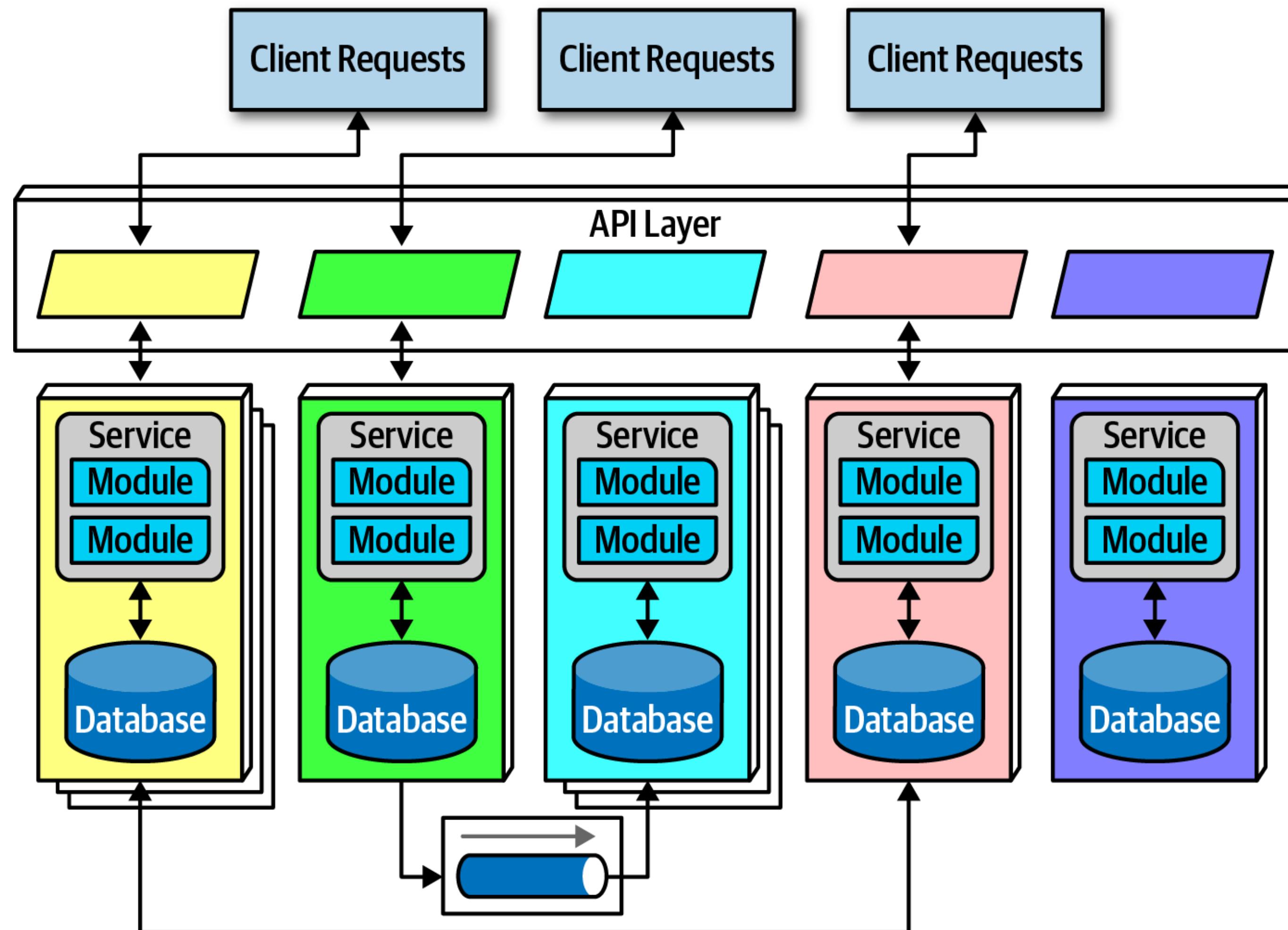
event-driven



space-based

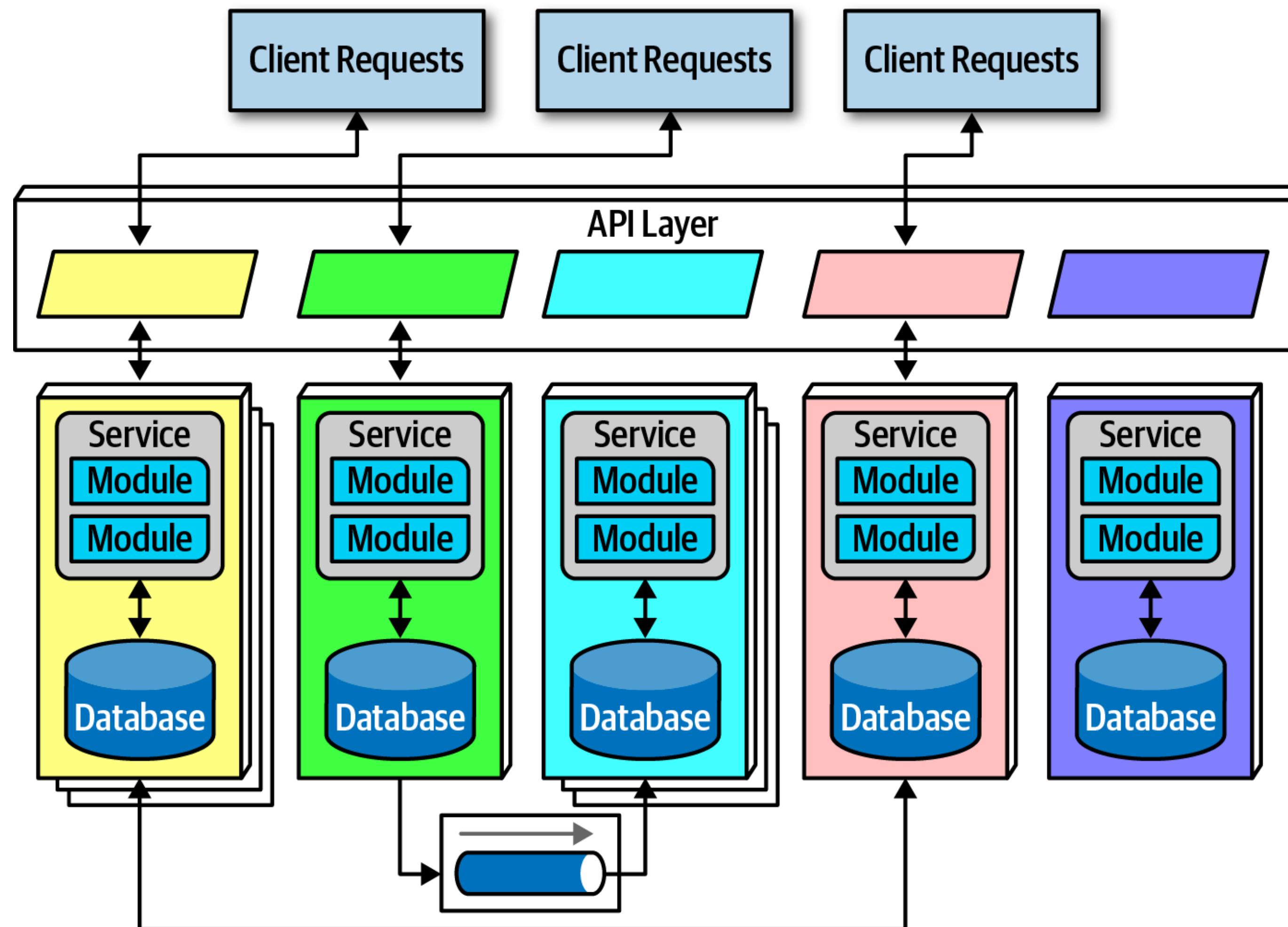
	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★	★★★★★	★★★★★	★★★★	★★★
testability	★★	★★★	★★★★★	★★★★★	★★★	★
performance	★★★★★	★★★★★	★★	★★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★★★	★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★
evolvability	★	★★★	★★★★★	★★★★★	★★★★★	★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★

# microservices

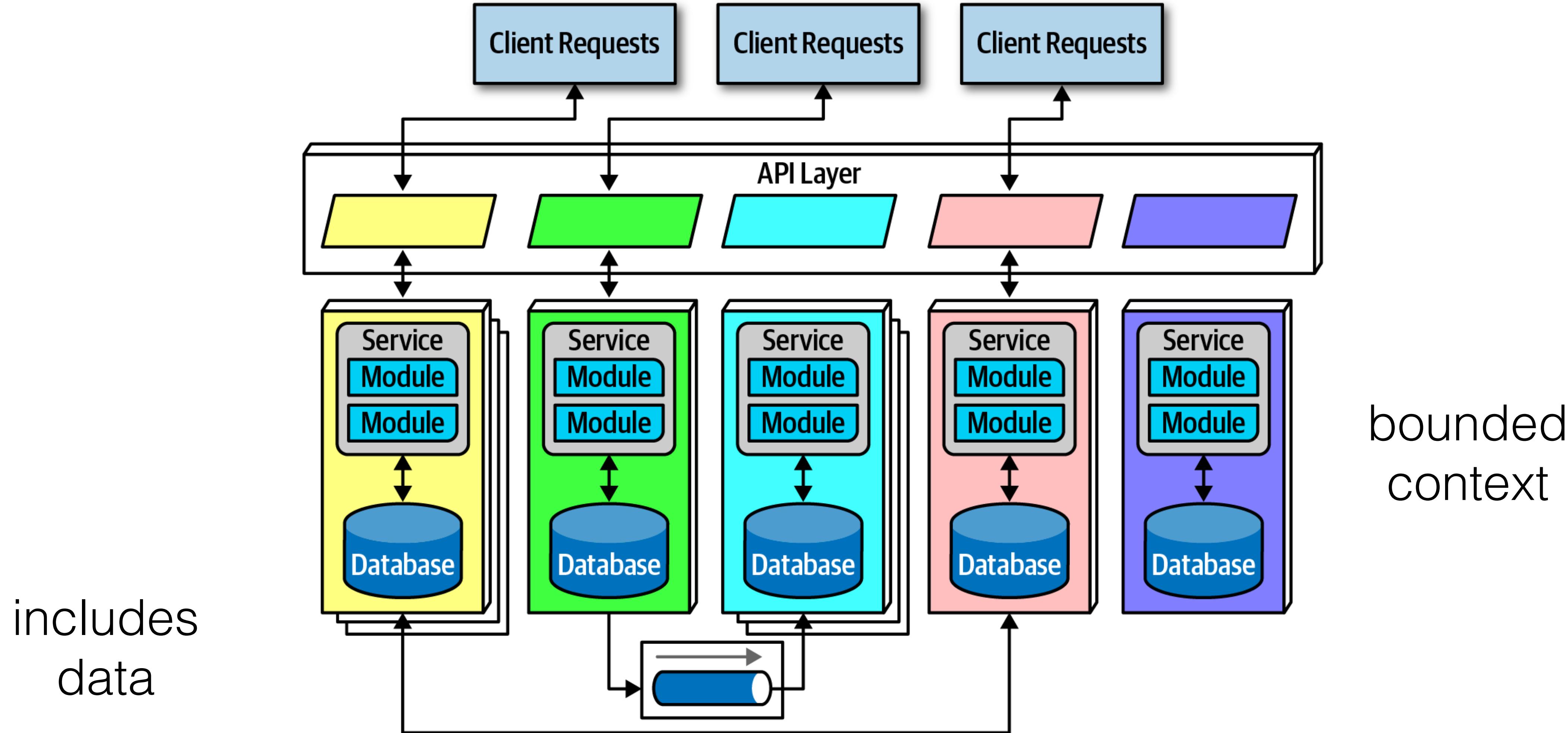


# microservices

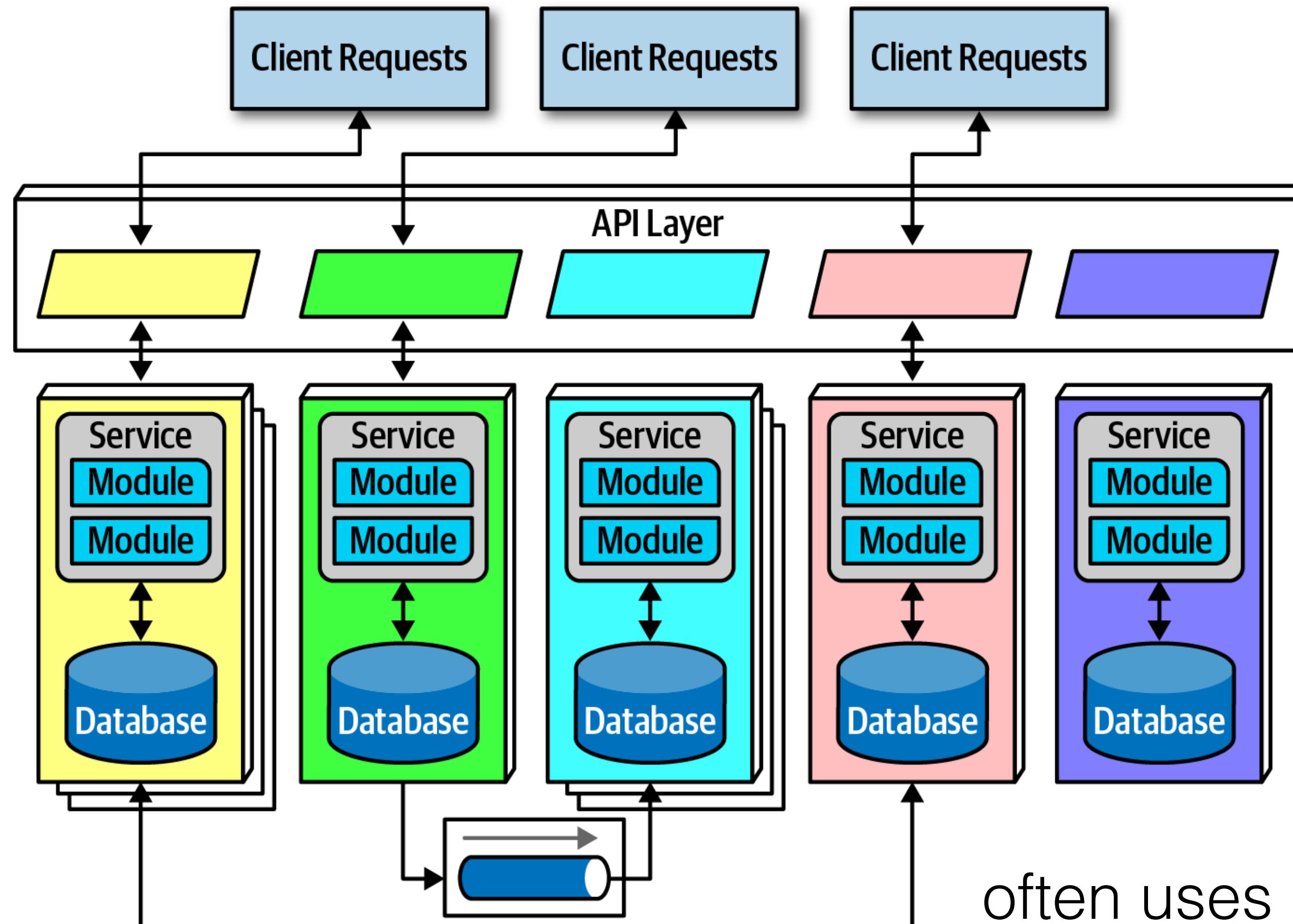
distributed  
architecture



# microservices

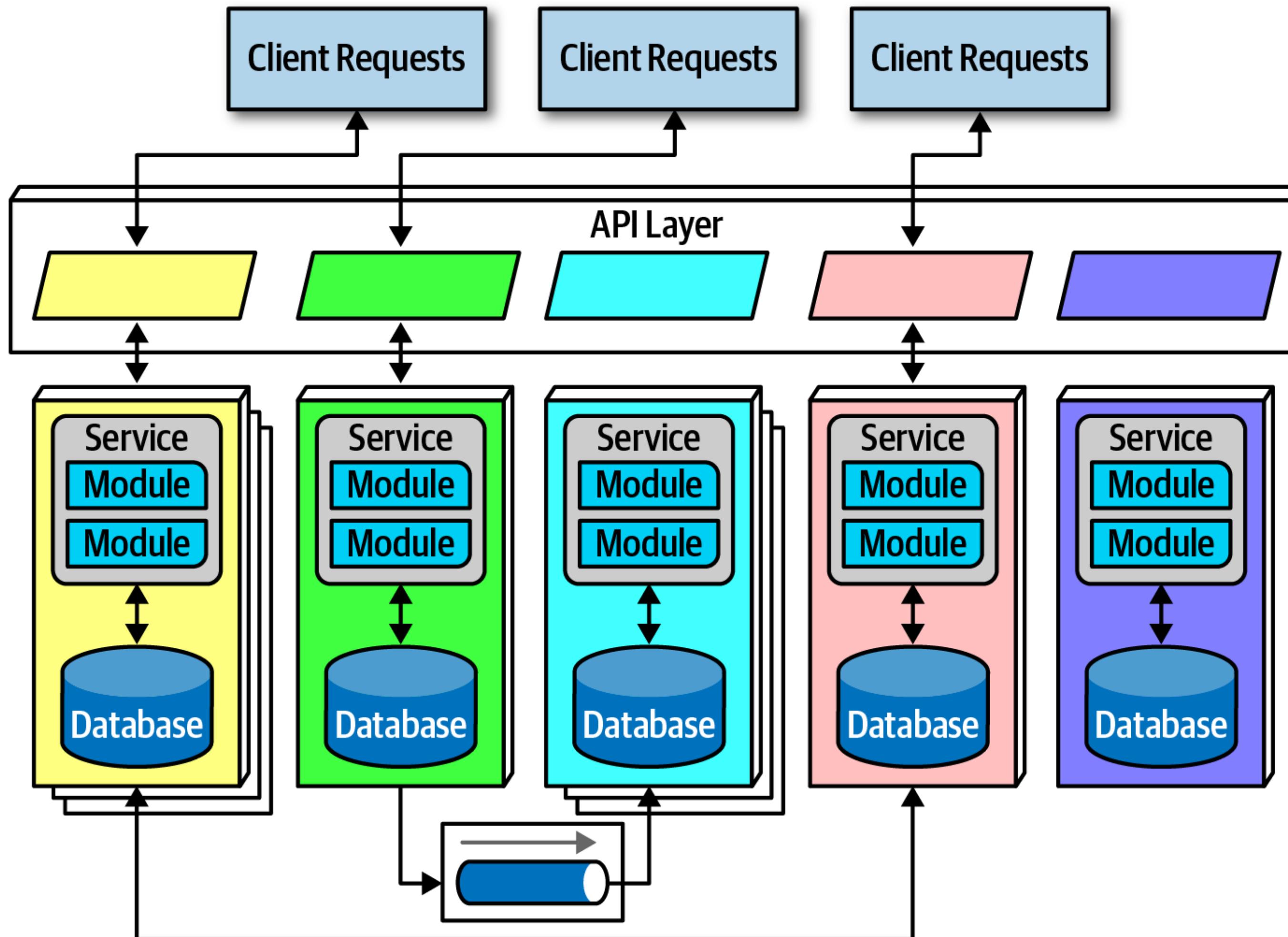


# microservices



often uses  
events

# suitability: microservices ?



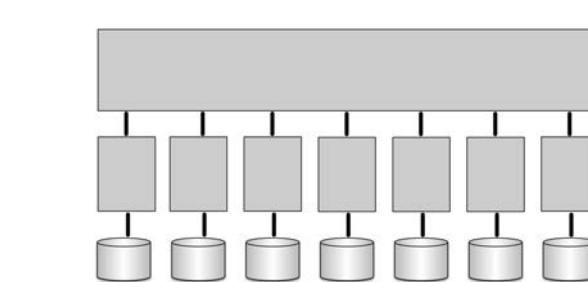
# Going Going Gone!



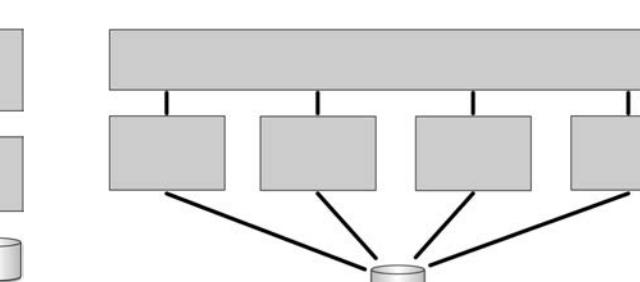
layered monolith



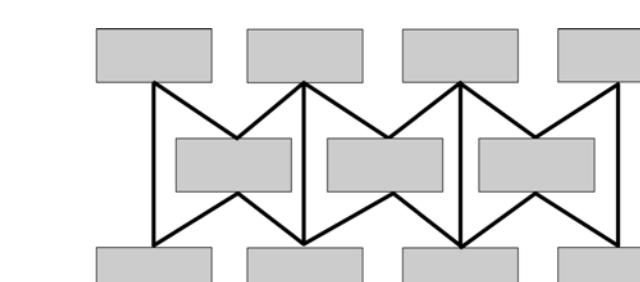
microkernel



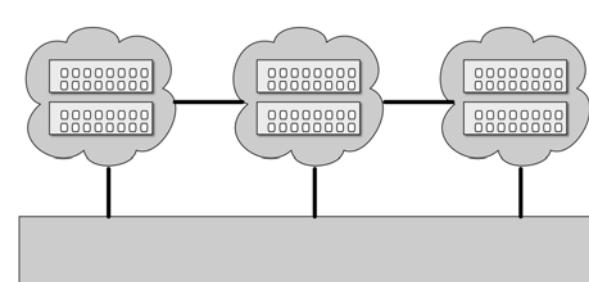
microservices



service-based



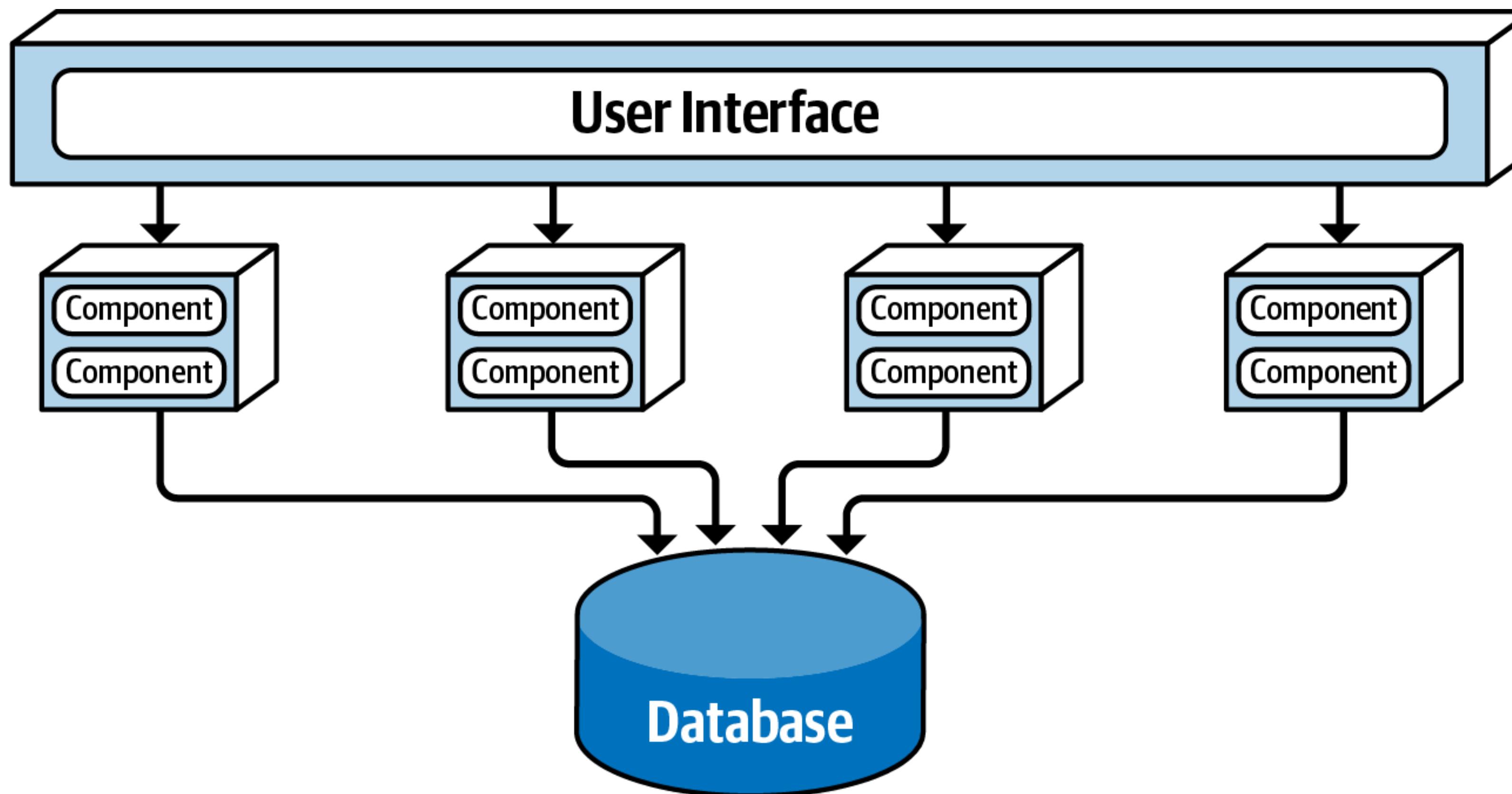
event-driven



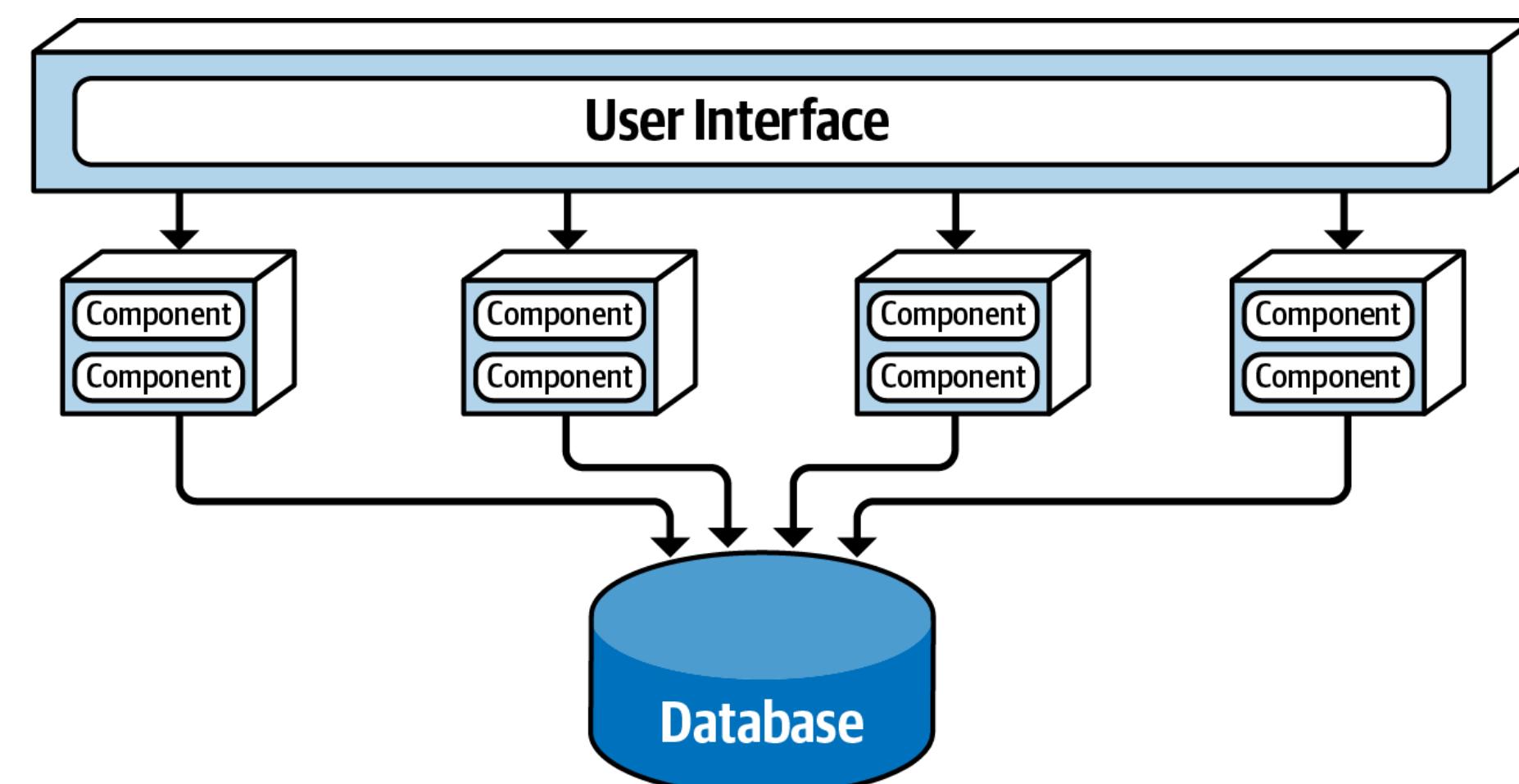
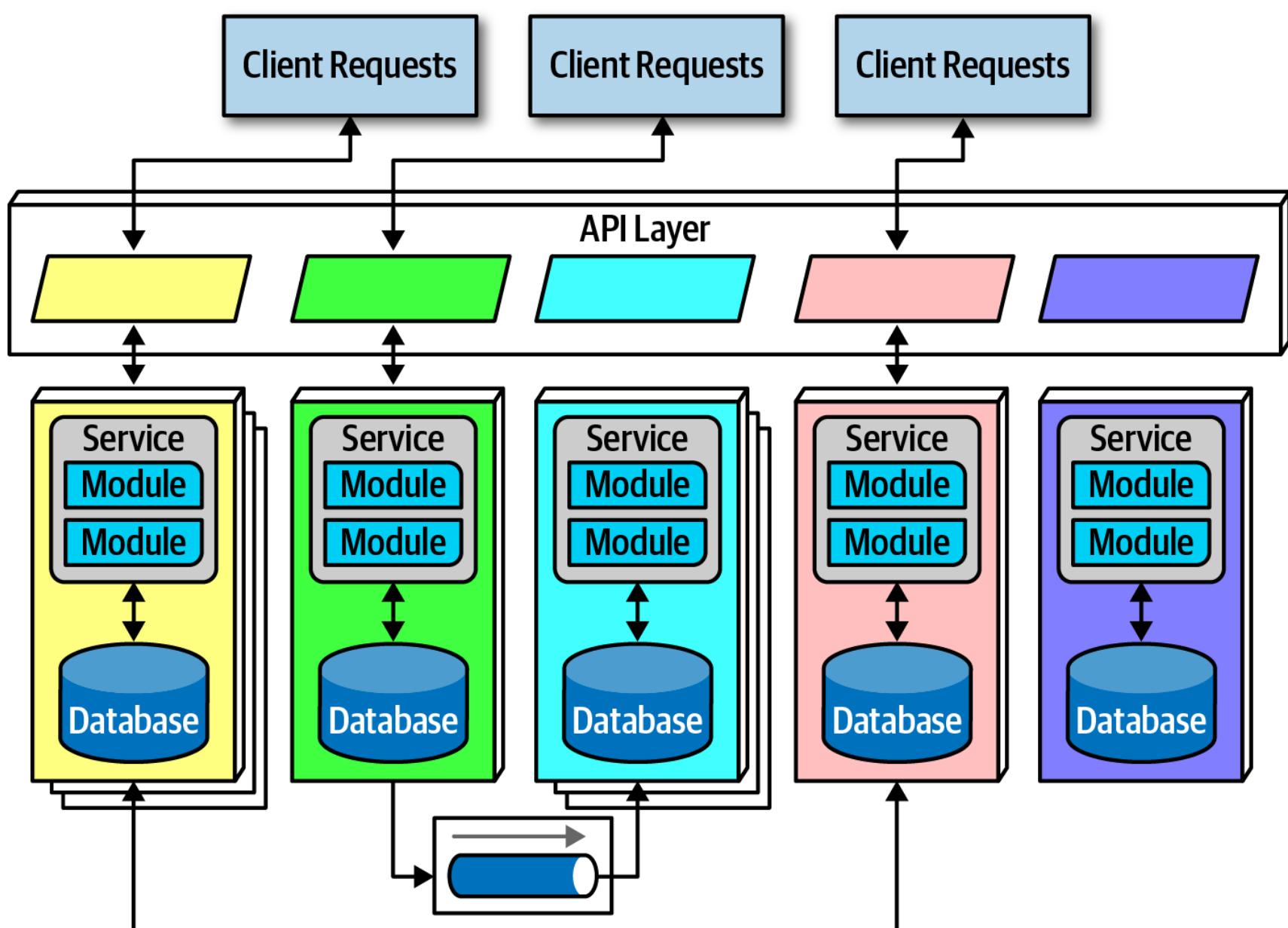
space-based

	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★	★★★★★	★★★★★	★★★★	★★★
testability	★★	★★★	★★★★★	★★★★★	★★★	★
performance	★★★★★	★★★★★	★★	★★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★★★	★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★
evolvability	★	★★★	★★★★★	★★★★★	★★★★★	★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★

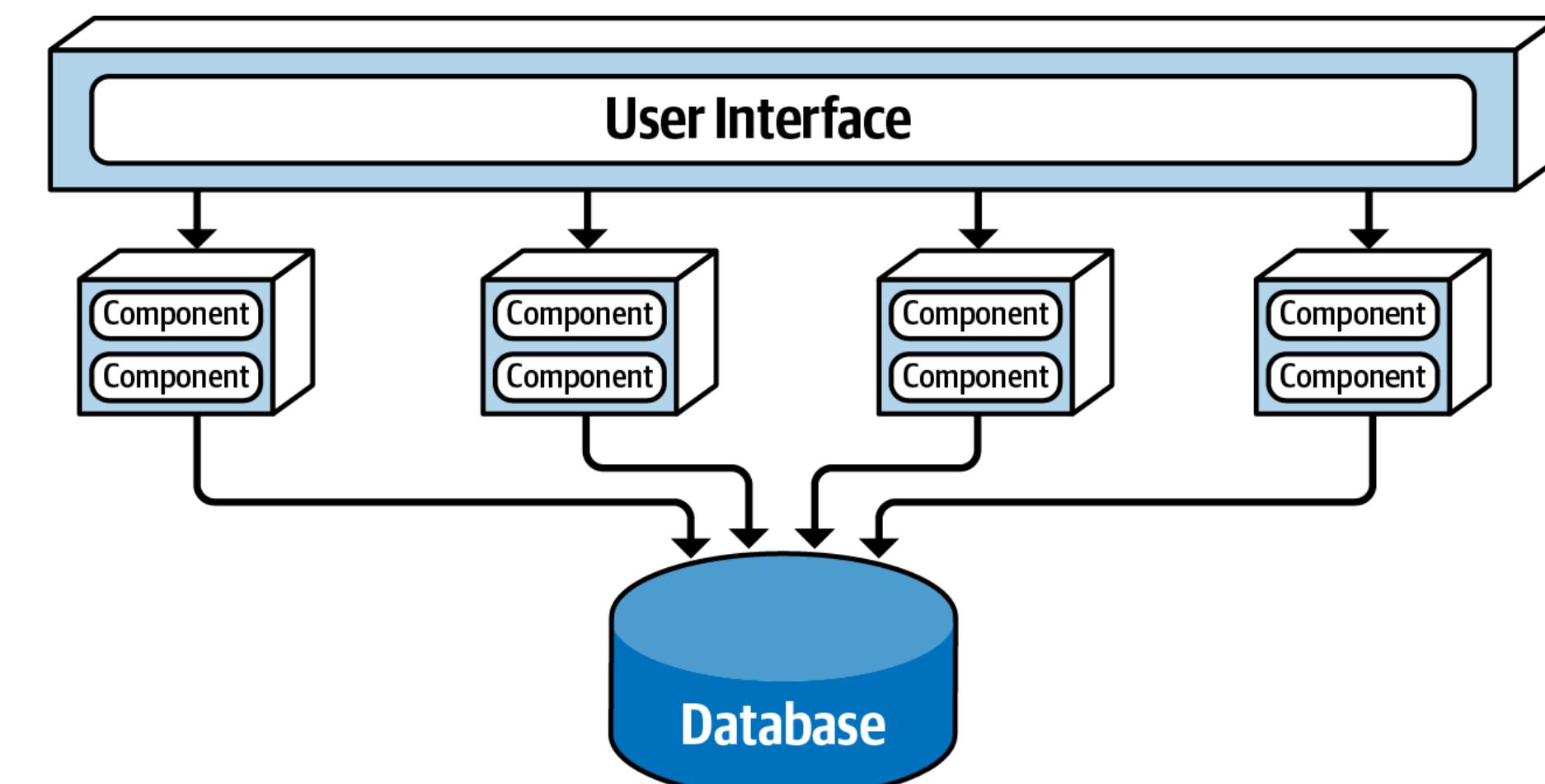
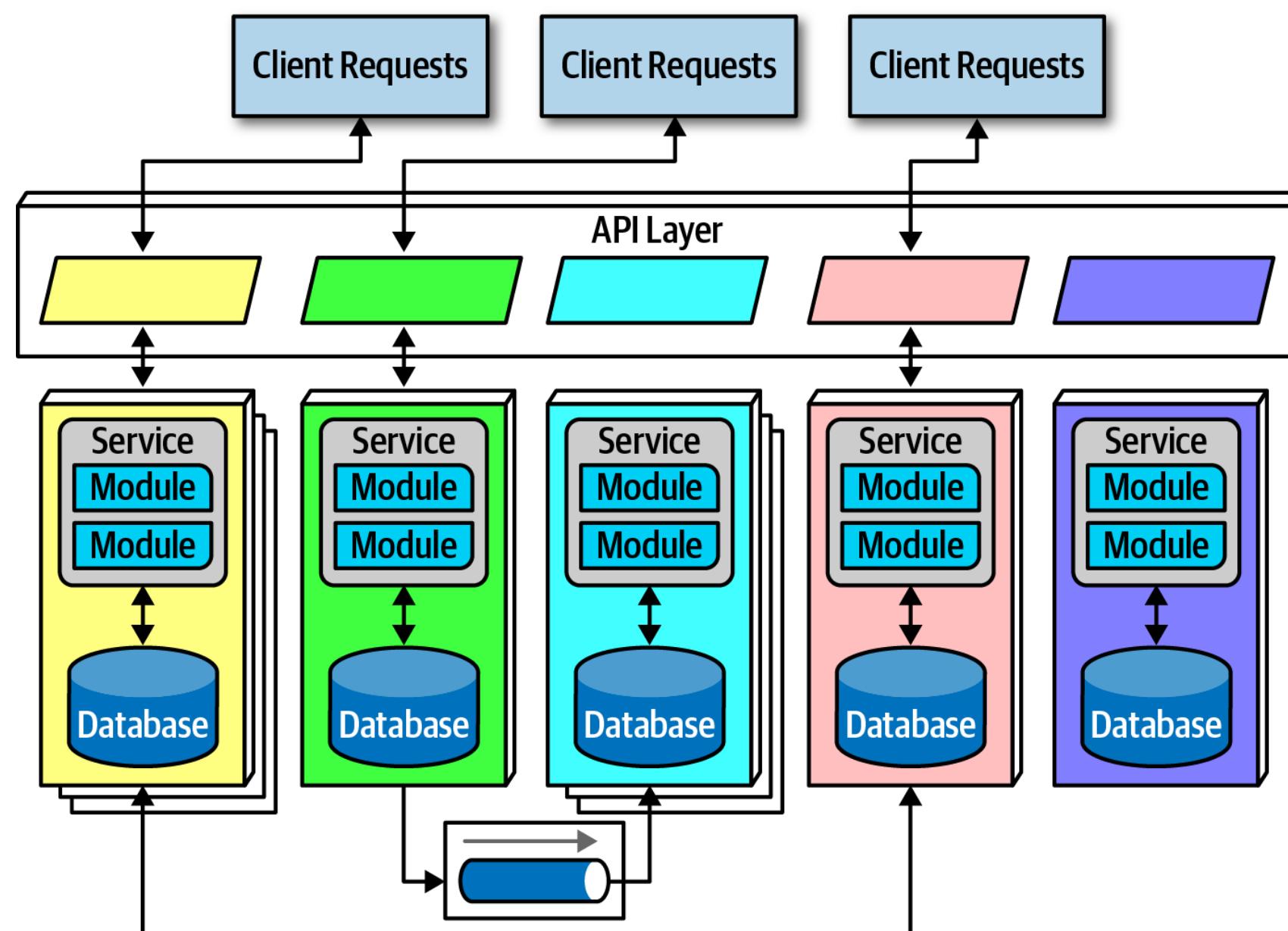
# service-based architecture



# microservices vs service-based

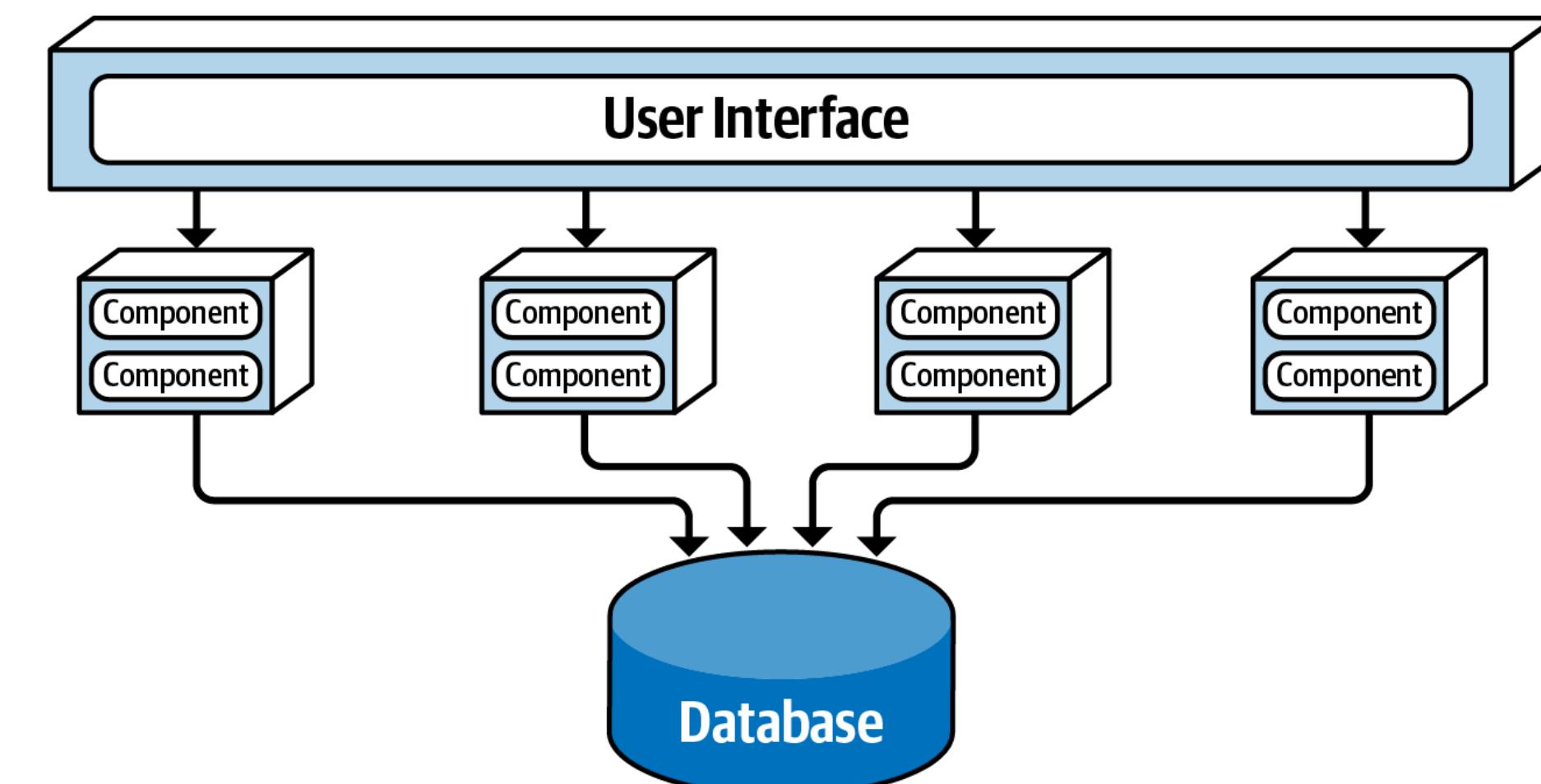
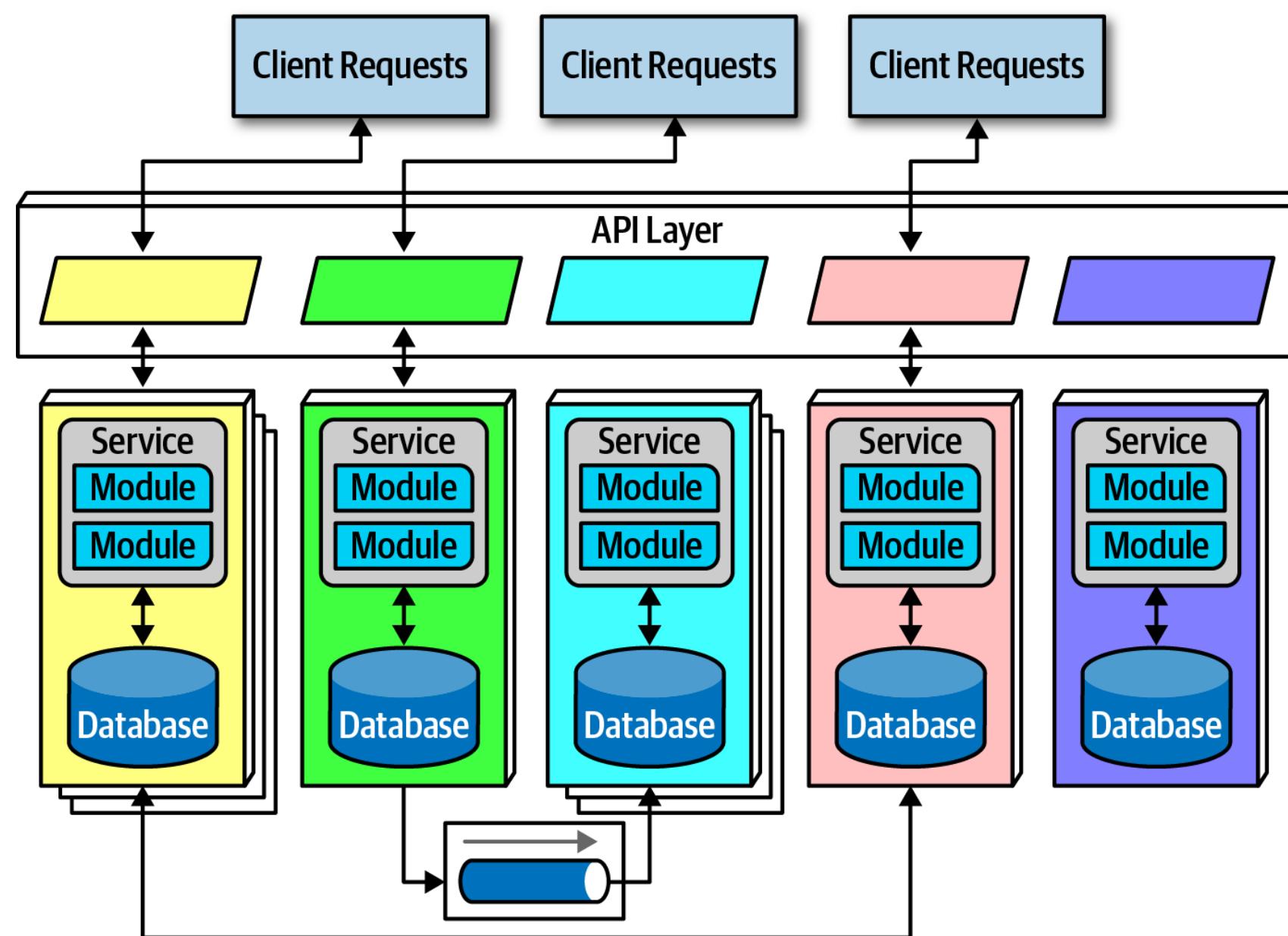


# microservices vs service-based



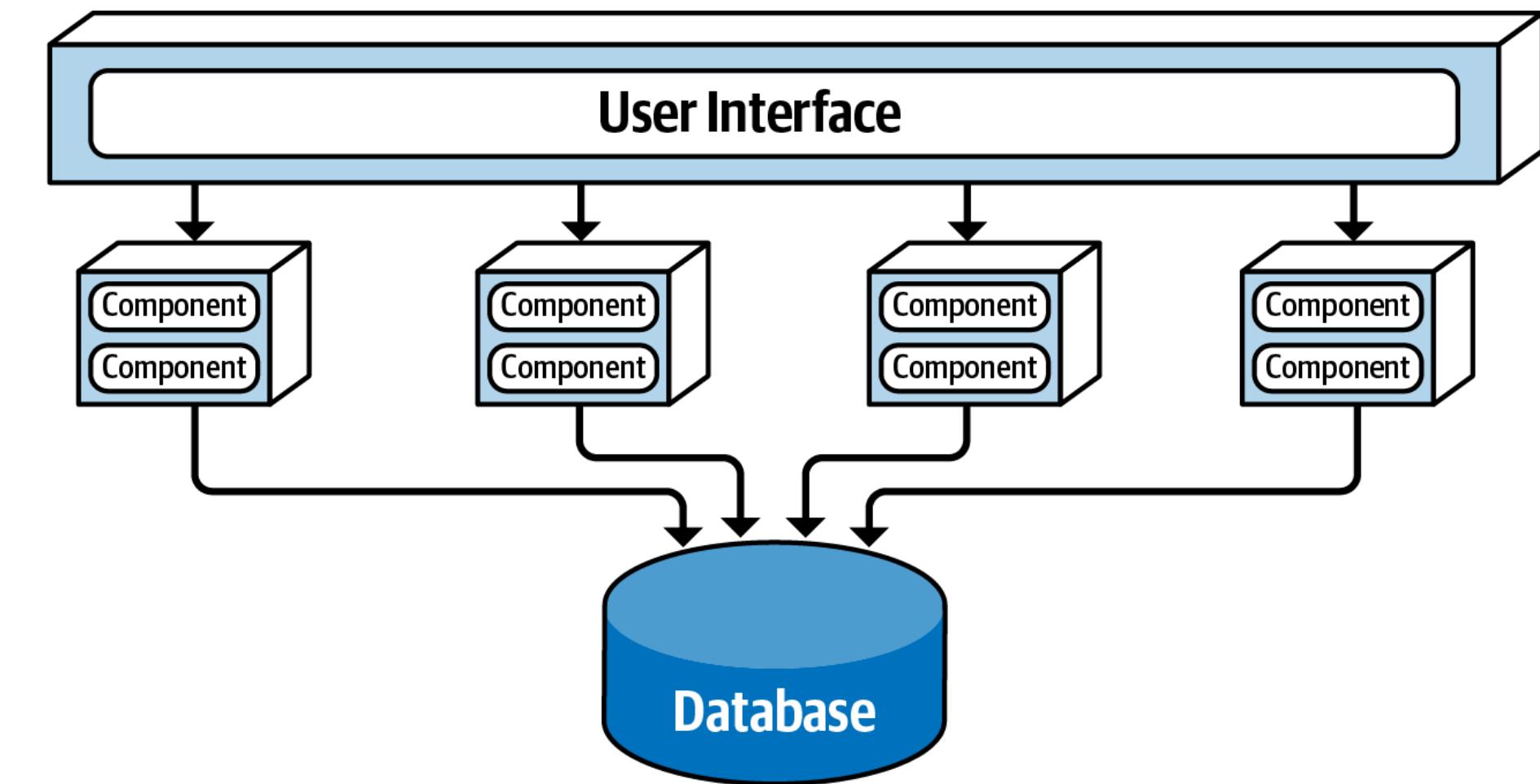
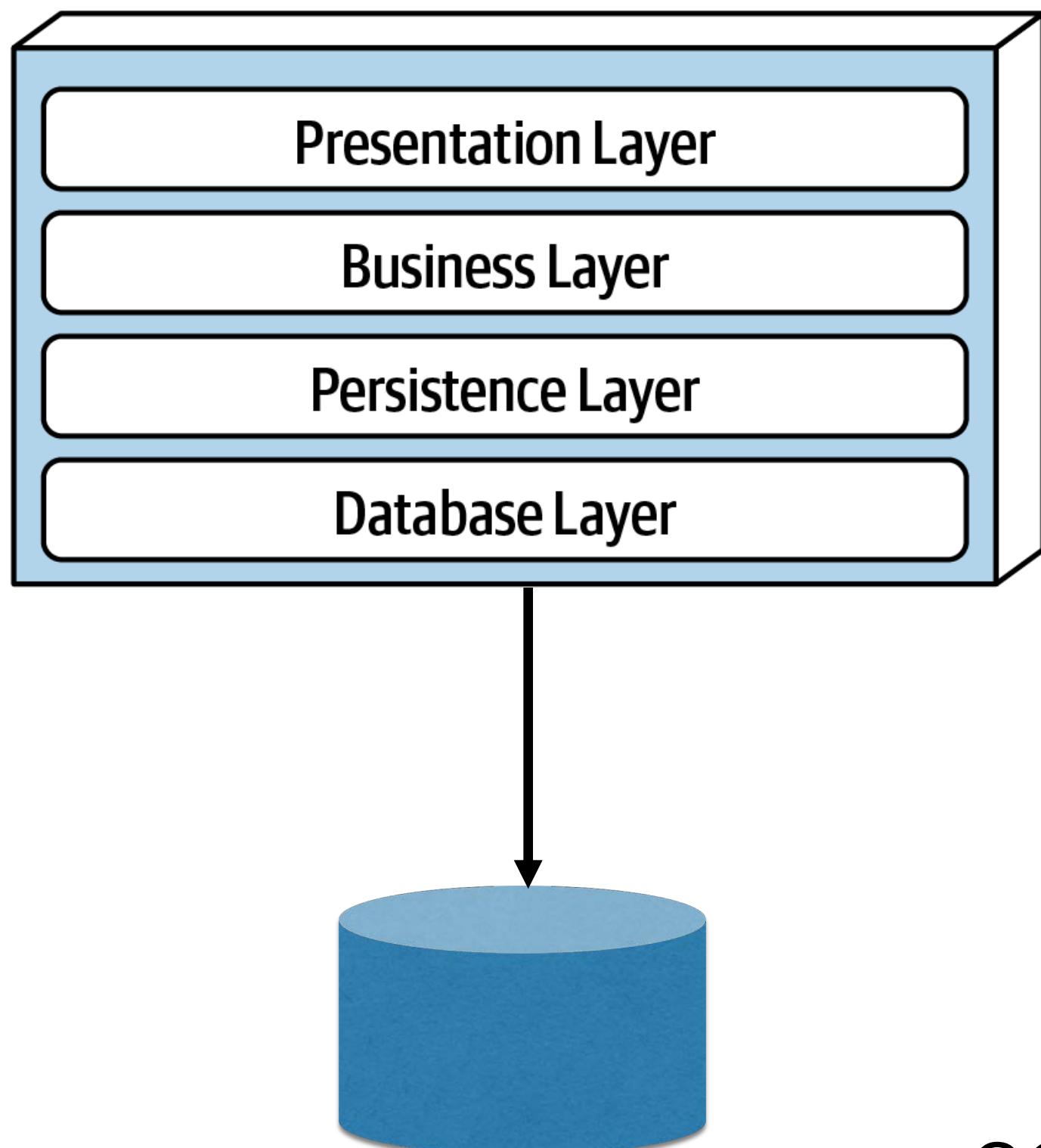
database(s)

# microservices vs service-based



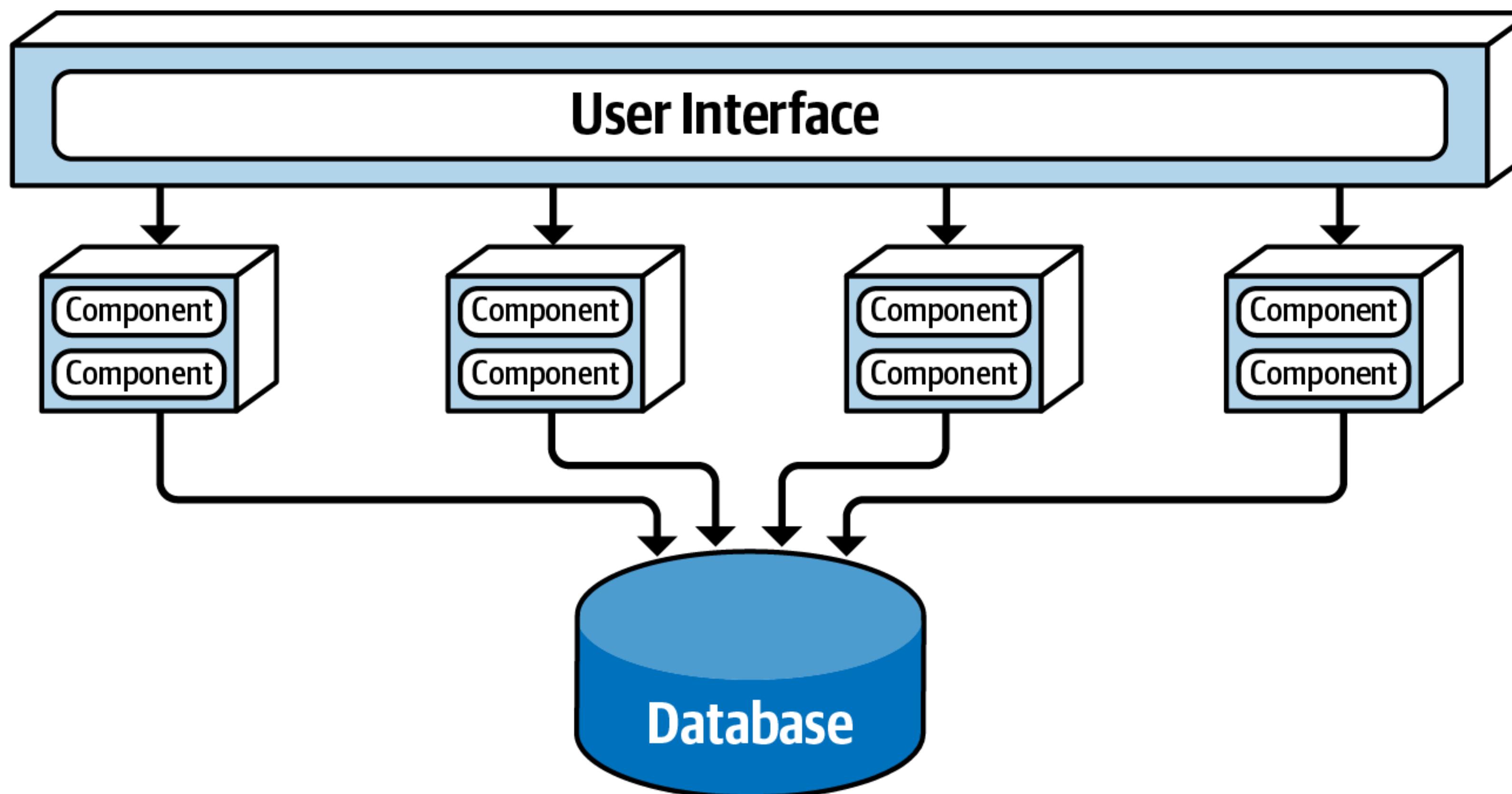
service granularity

# monolith => service-based



common migration  
target

# suitability: service-based ?



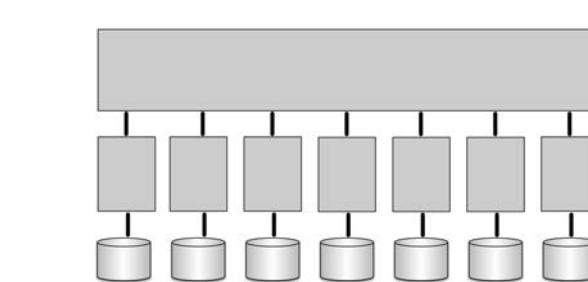
# Going Going Gone!



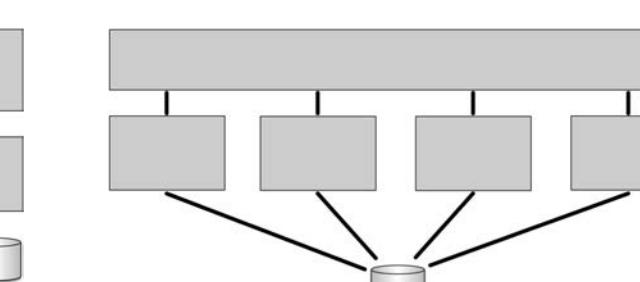
layered monolith



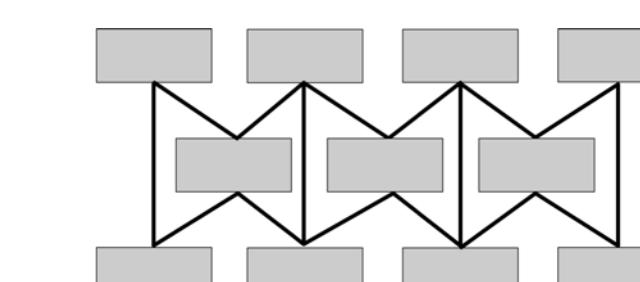
microkernel



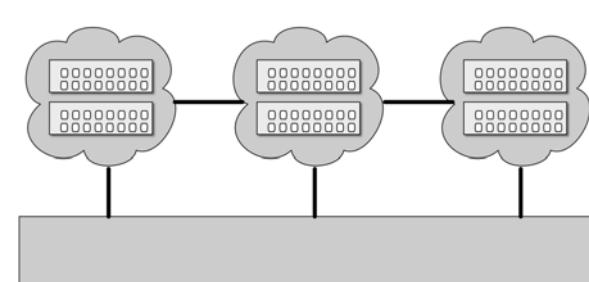
microservices



service-based



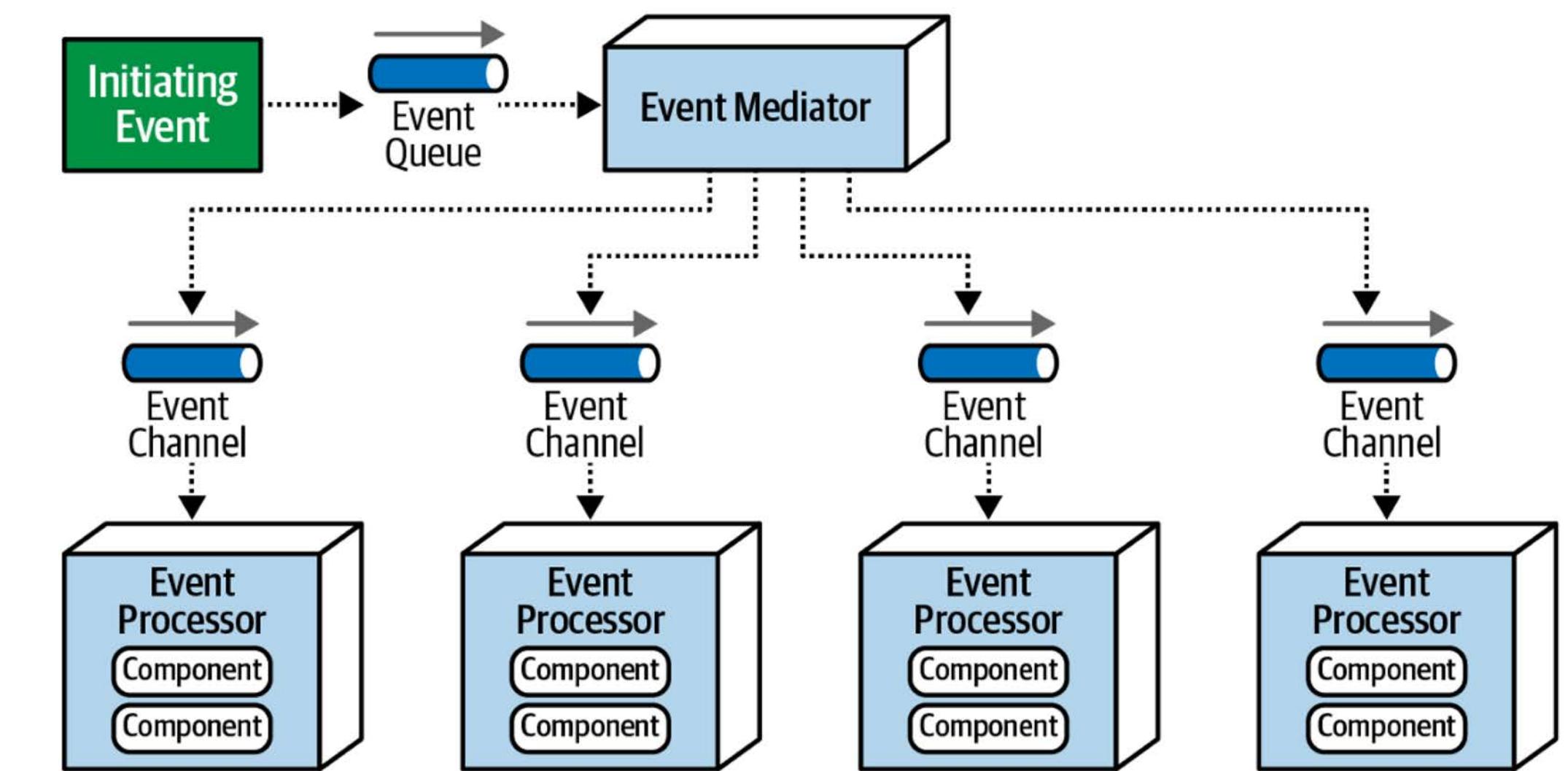
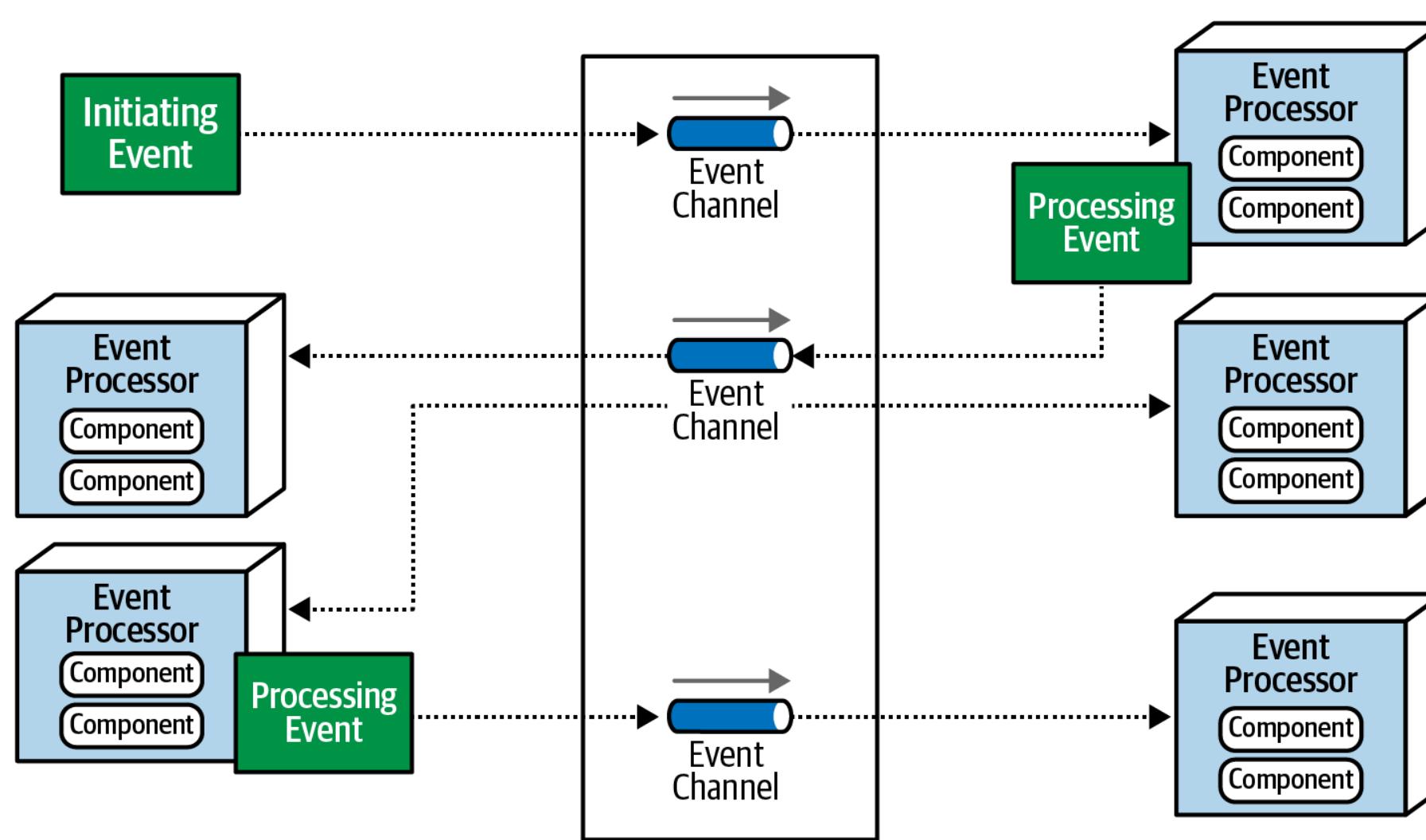
event-driven



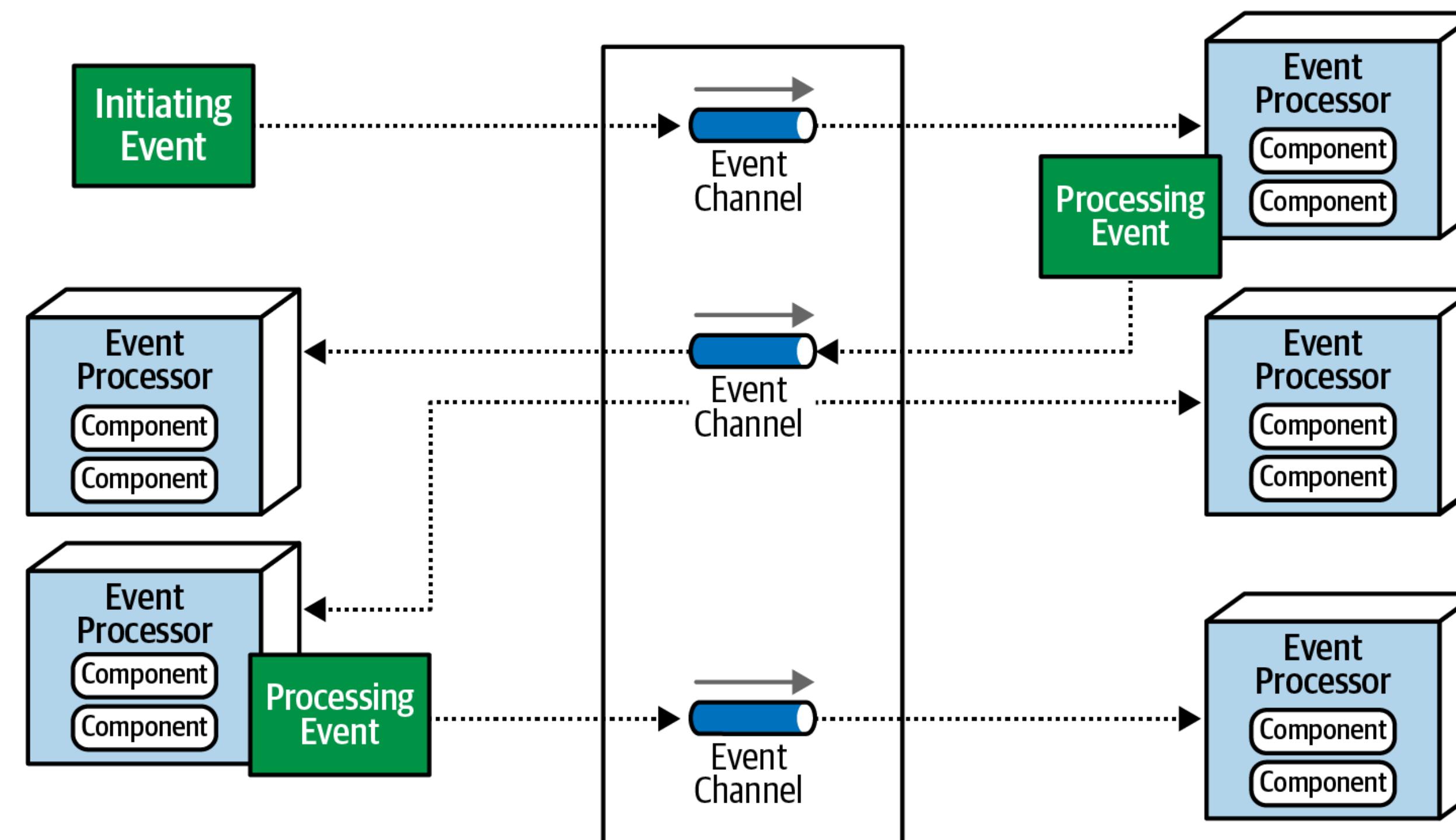
space-based

	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	1 star	4 stars	5 stars	4 stars	4 stars	4 stars
deployment	1 star	4 stars	5 stars	4 stars	4 stars	4 stars
testability	2 stars	4 stars	5 stars	4 stars	2 stars	1 star
performance	4 stars	4 stars	2 stars	3 stars	5 stars	5 stars
scalability	1 star	1 star	5 stars	3 stars	4 stars	5 stars
elasticity	1 star	1 star	4 stars	3 stars	3 stars	5 stars
simplicity	5 stars	5 stars	1 star	3 stars	1 star	1 star
fault-tolerance	1 star	1 star	5 stars	5 stars	5 stars	3 stars
evolvability	1 star	4 stars	5 stars	5 stars	5 stars	3 stars
total cost	5 stars	6 stars	1 star	5 stars	3 stars	2 stars

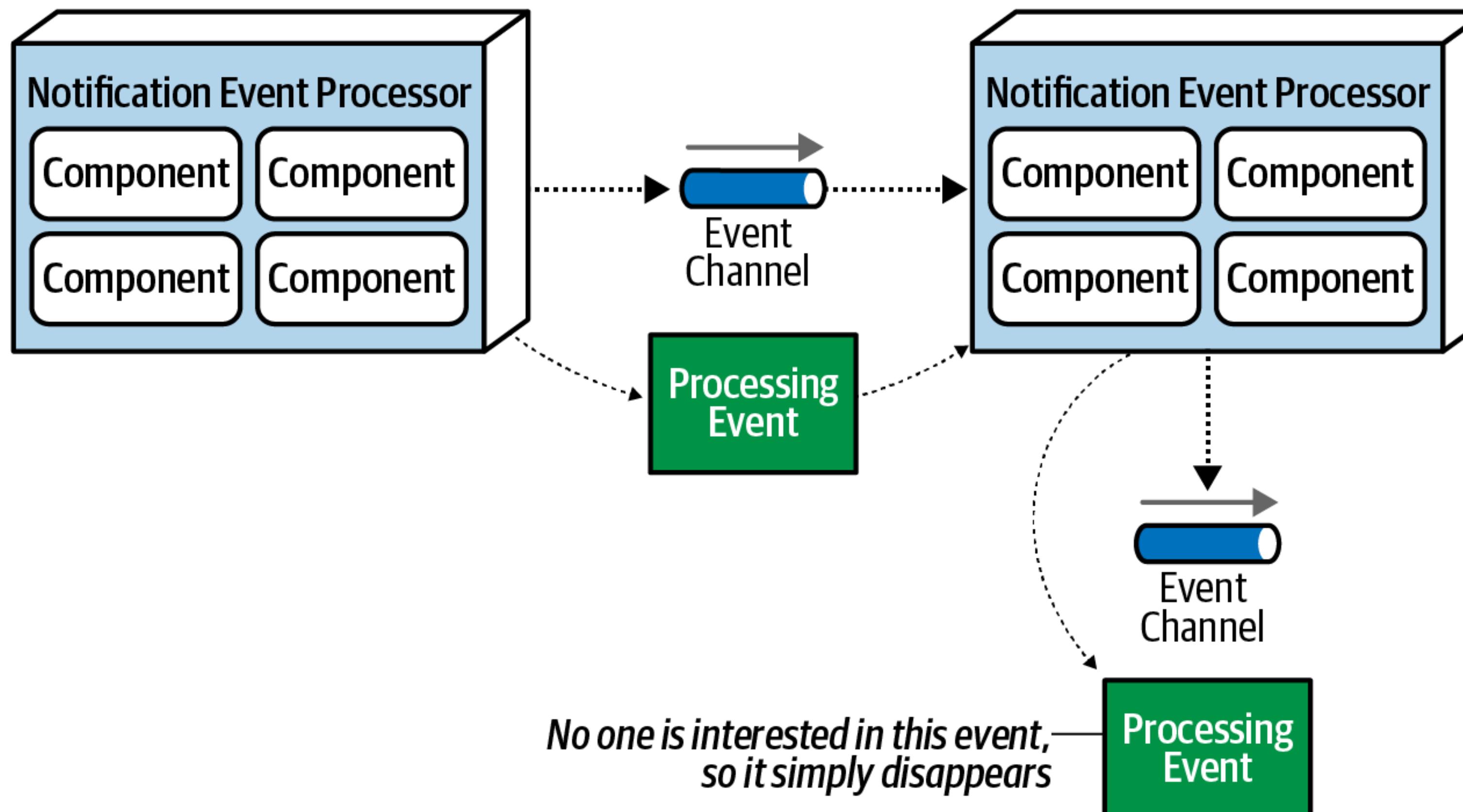
# event-driven architecture



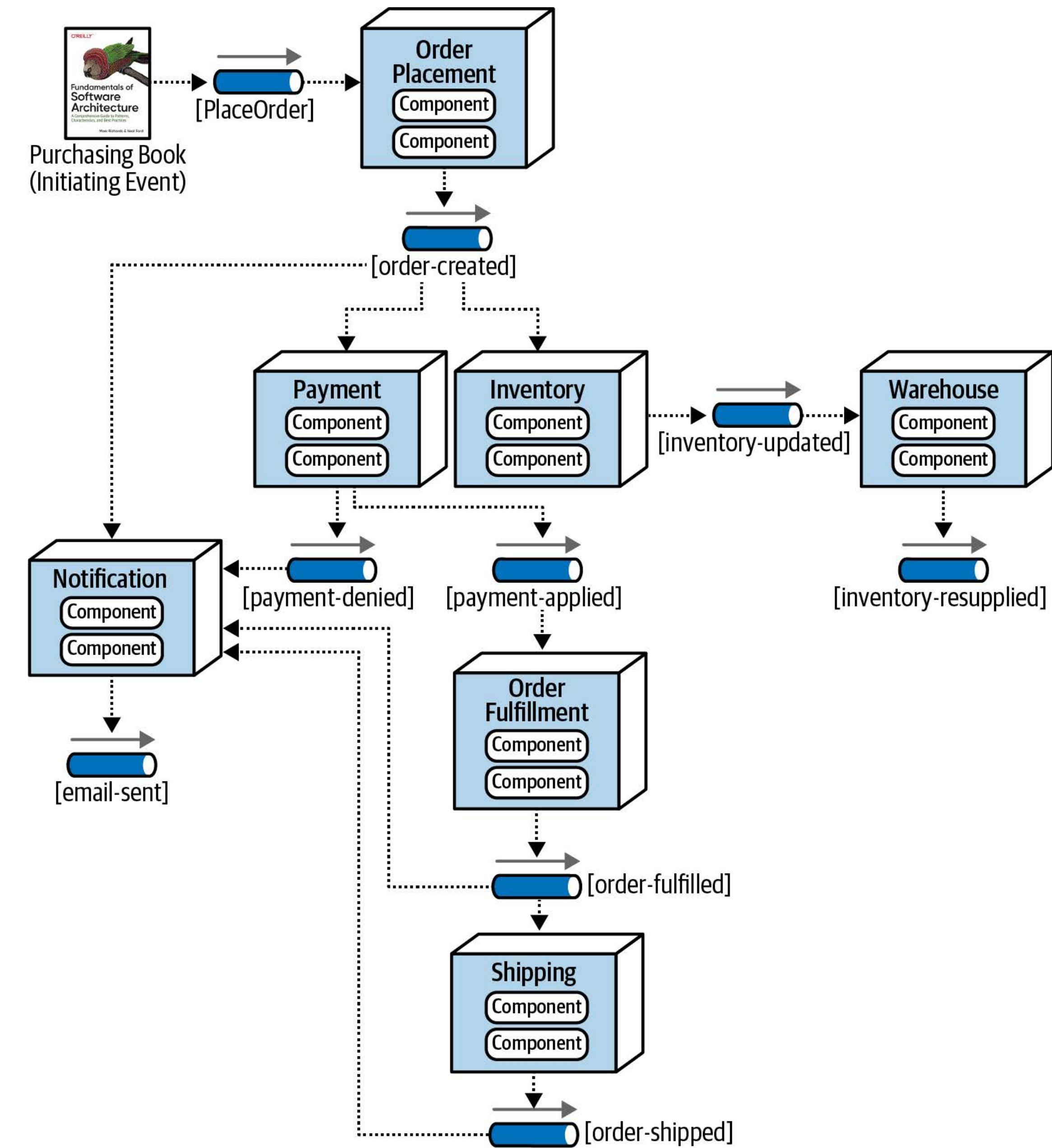
# EDA: broker



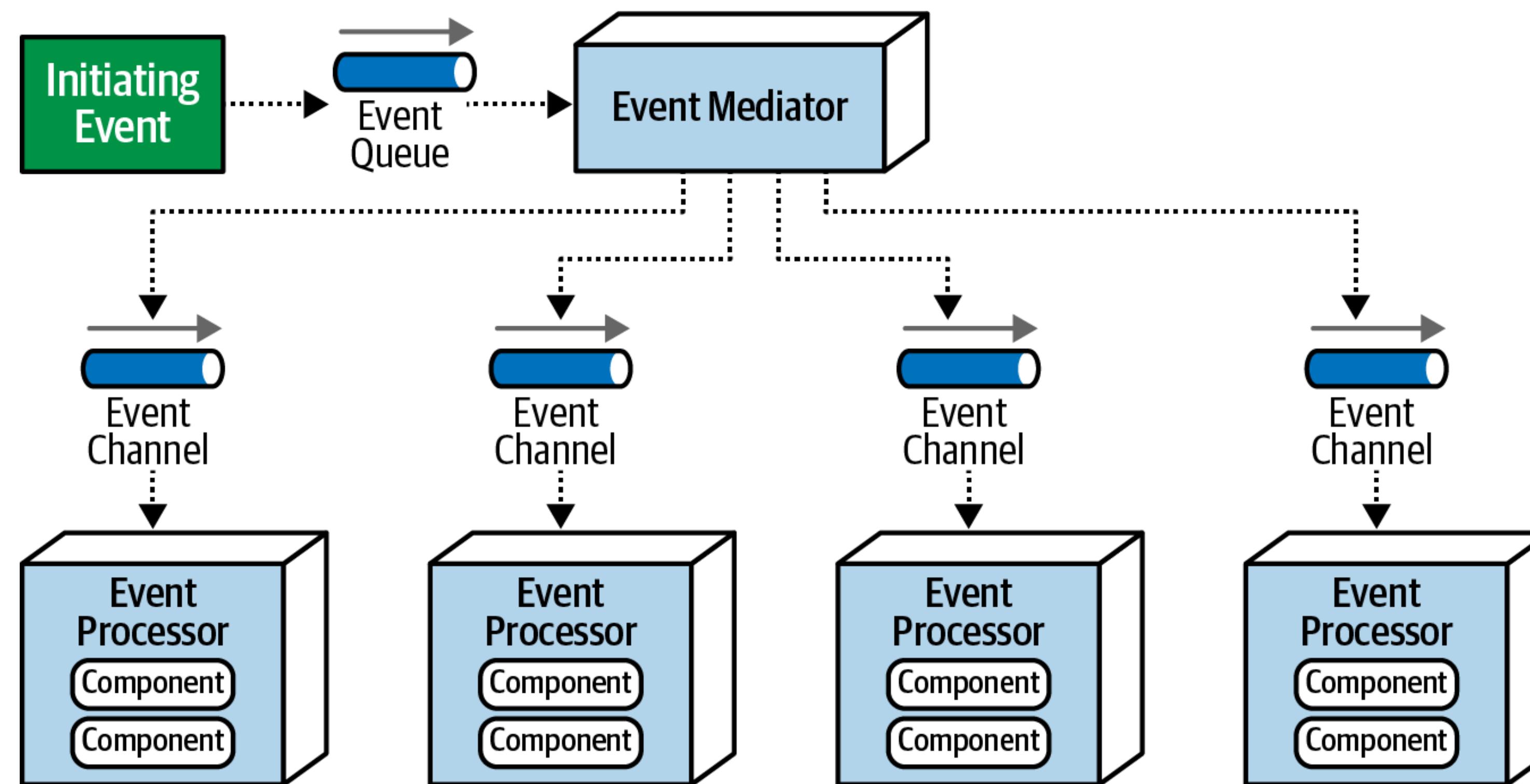
# event-driven architecture



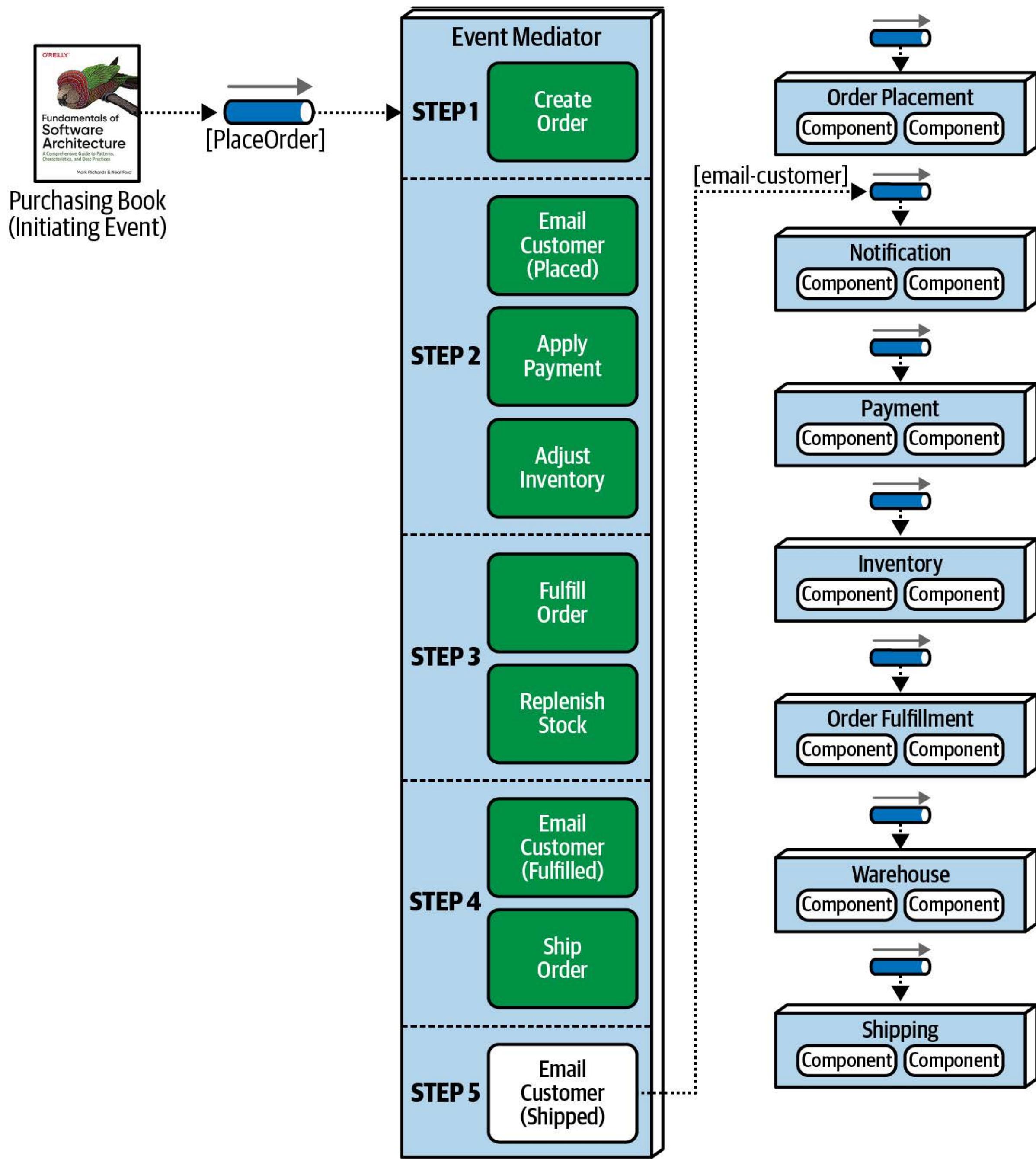
# EDA: broker



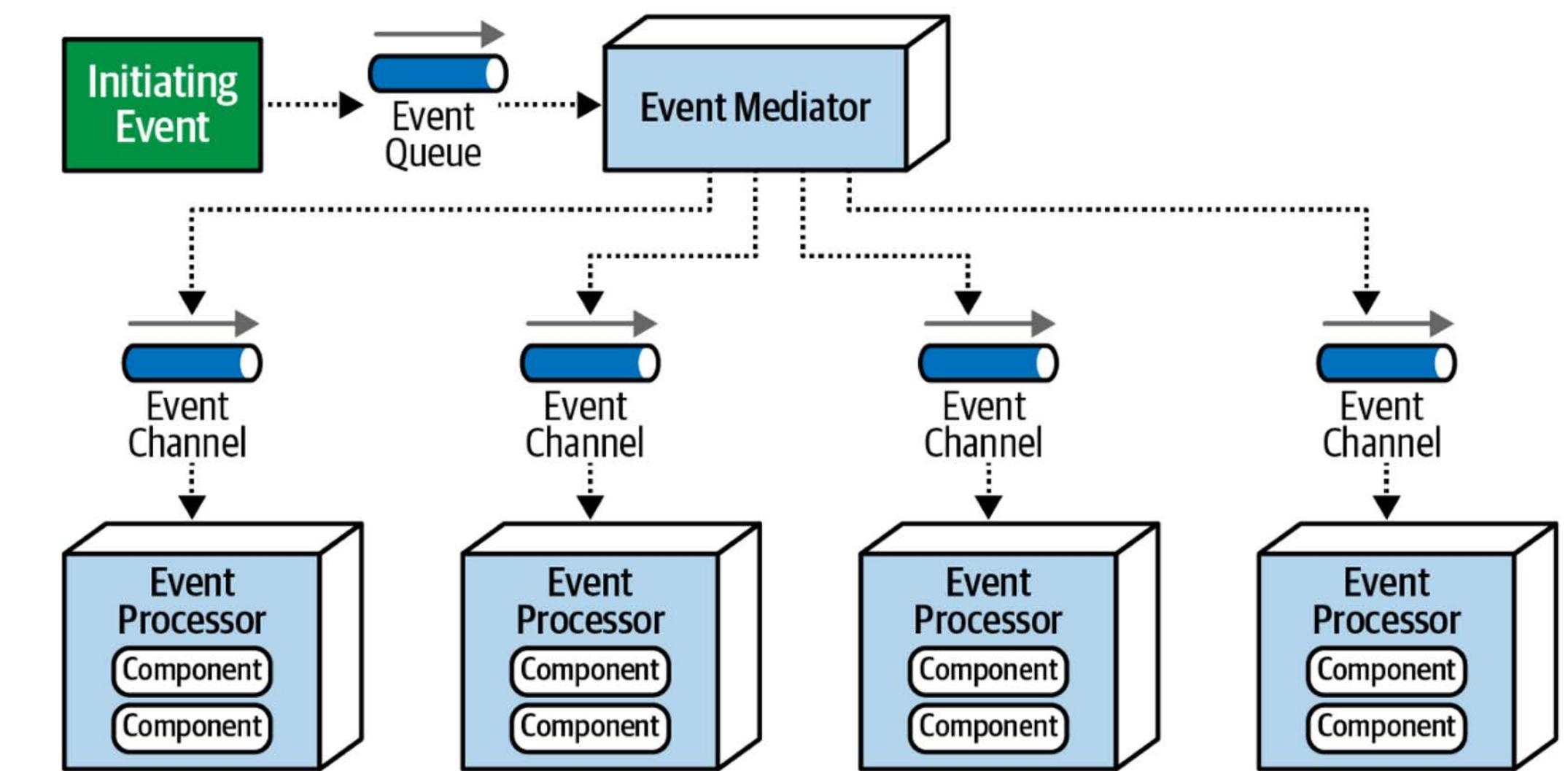
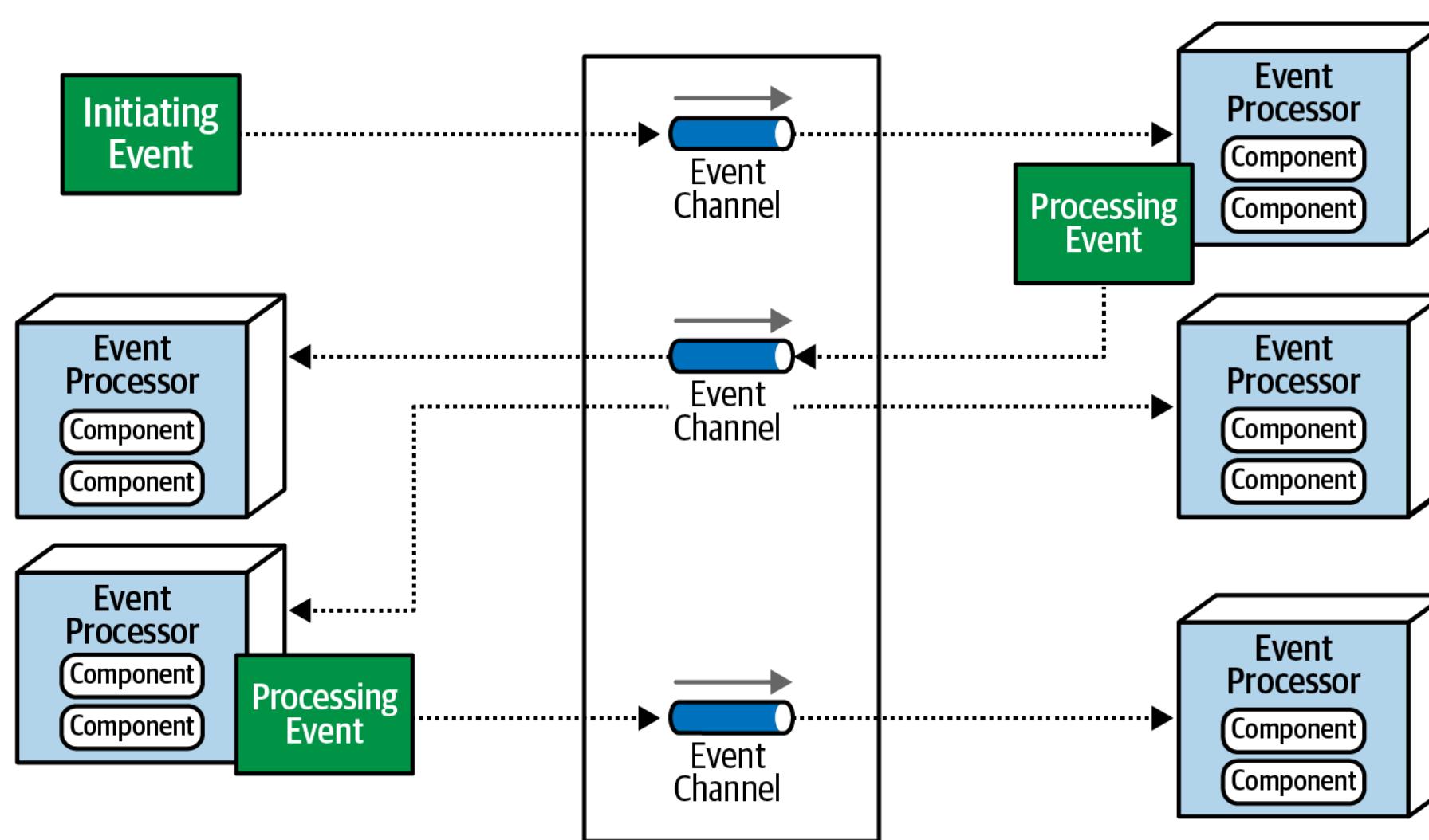
# EDA: mediator



# EDA: mediator



# suitability: EDA ?



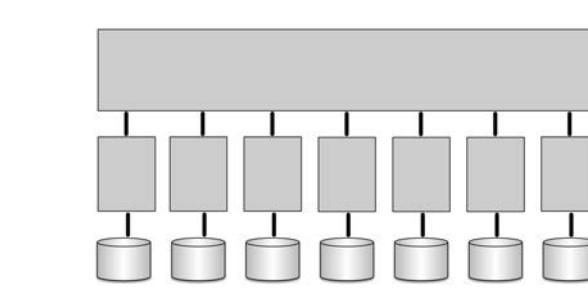
# Going Going Gone!



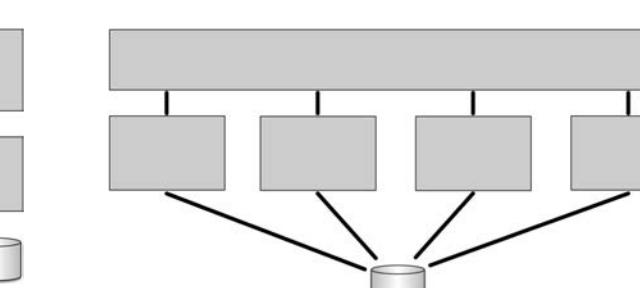
layered monolith



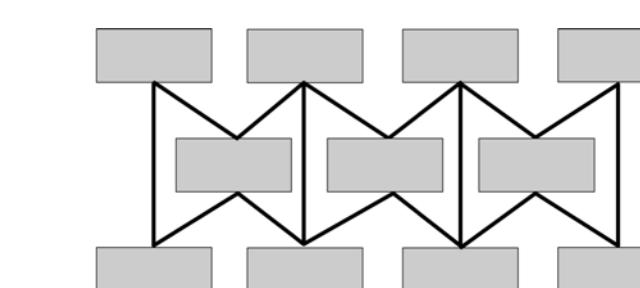
microkernel



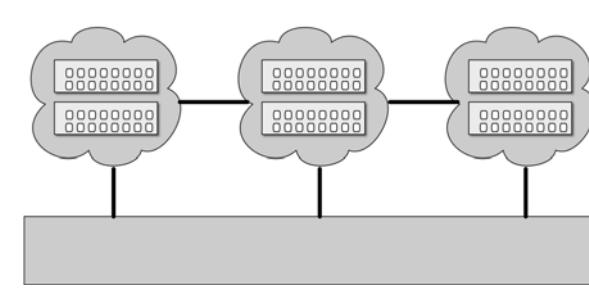
microservices



service-based



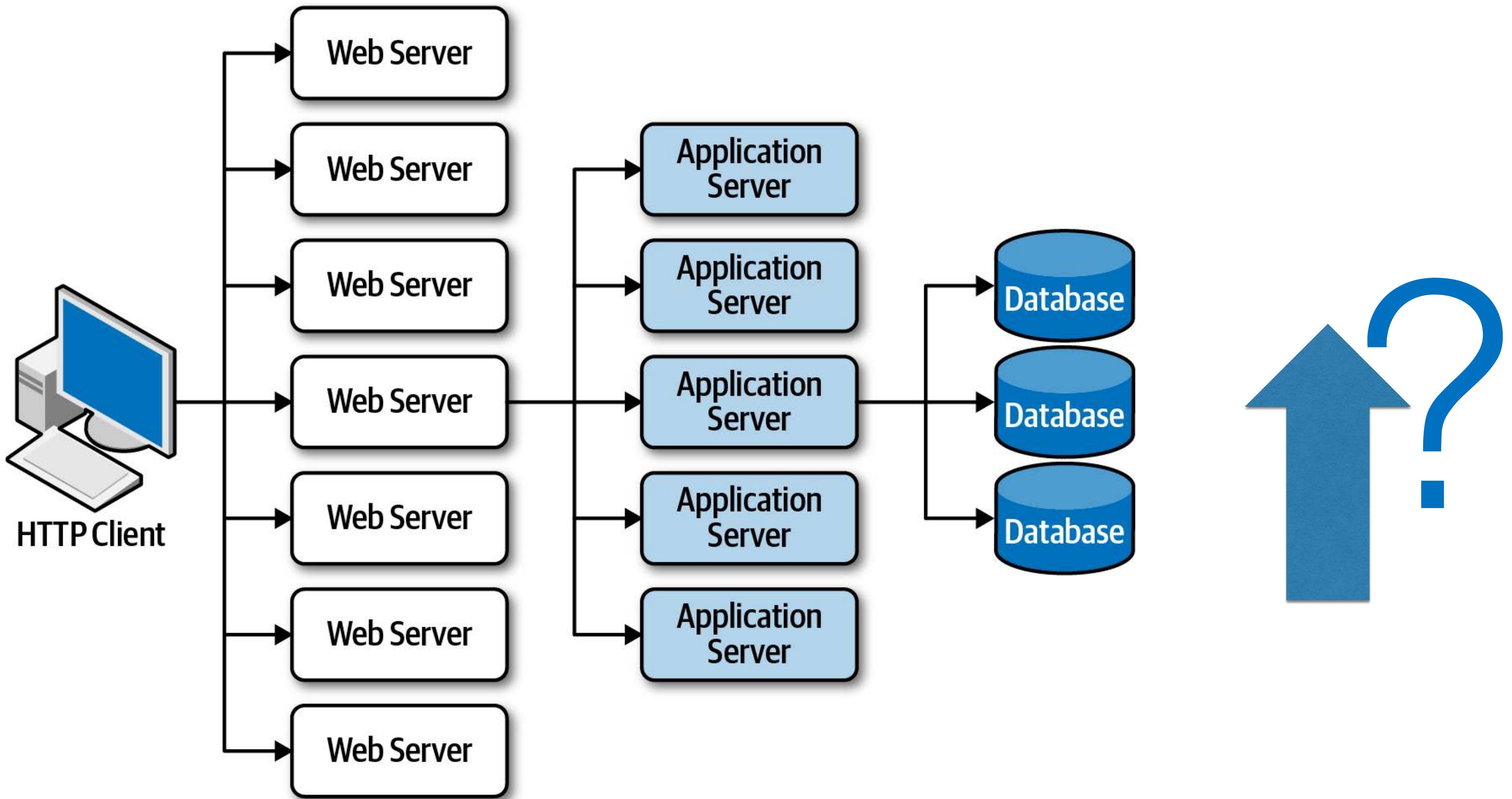
event-driven



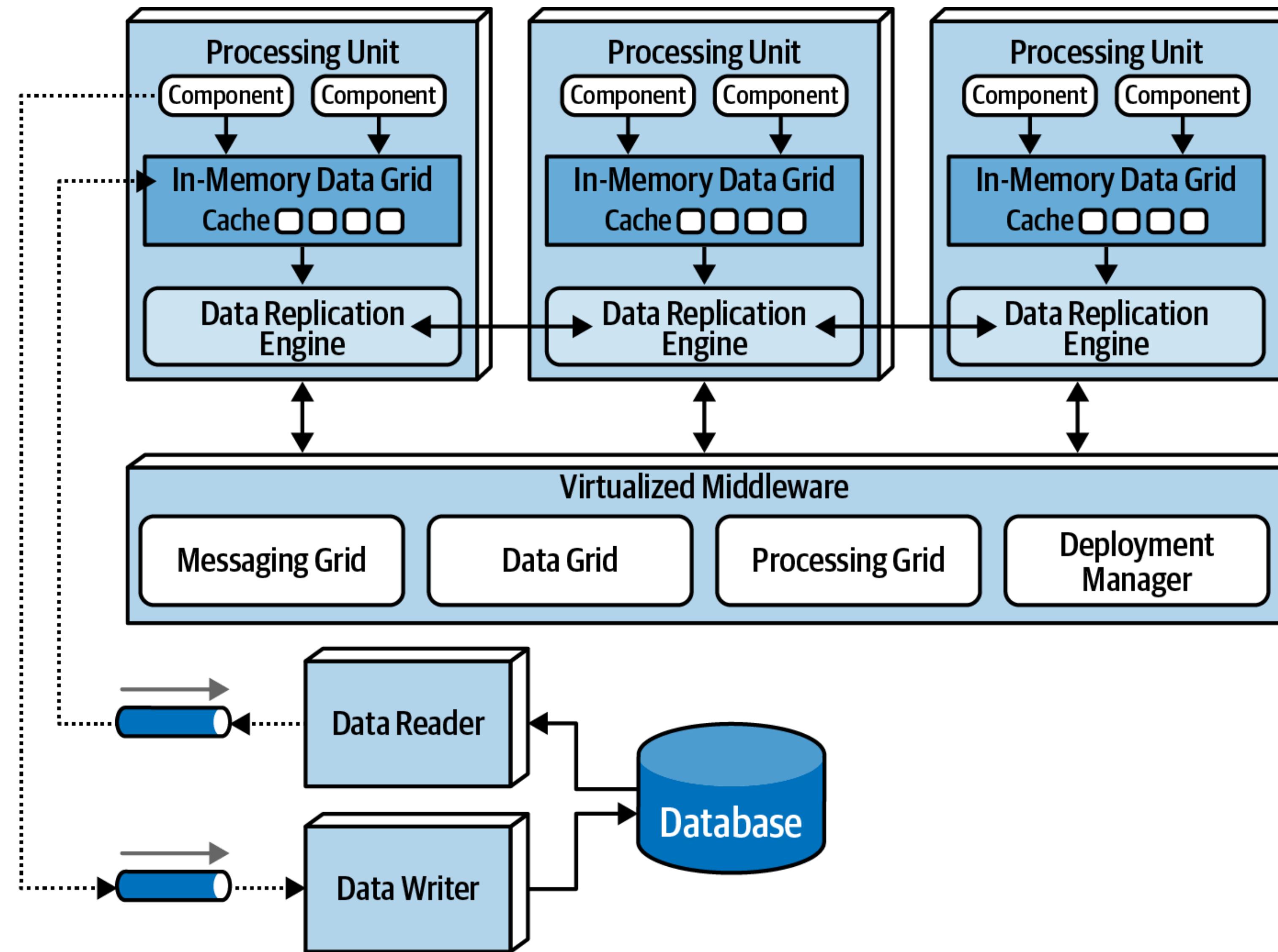
space-based

	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★	★★★★★	★★★★★	★★★★	★★★
testability	★★	★★★	★★★★★	★★★★★	★★★	★
performance	★★★★★	★★★★★	★★	★★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★★★	★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★
evolvability	★	★★★	★★★★★	★★★★★	★★★★★	★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★

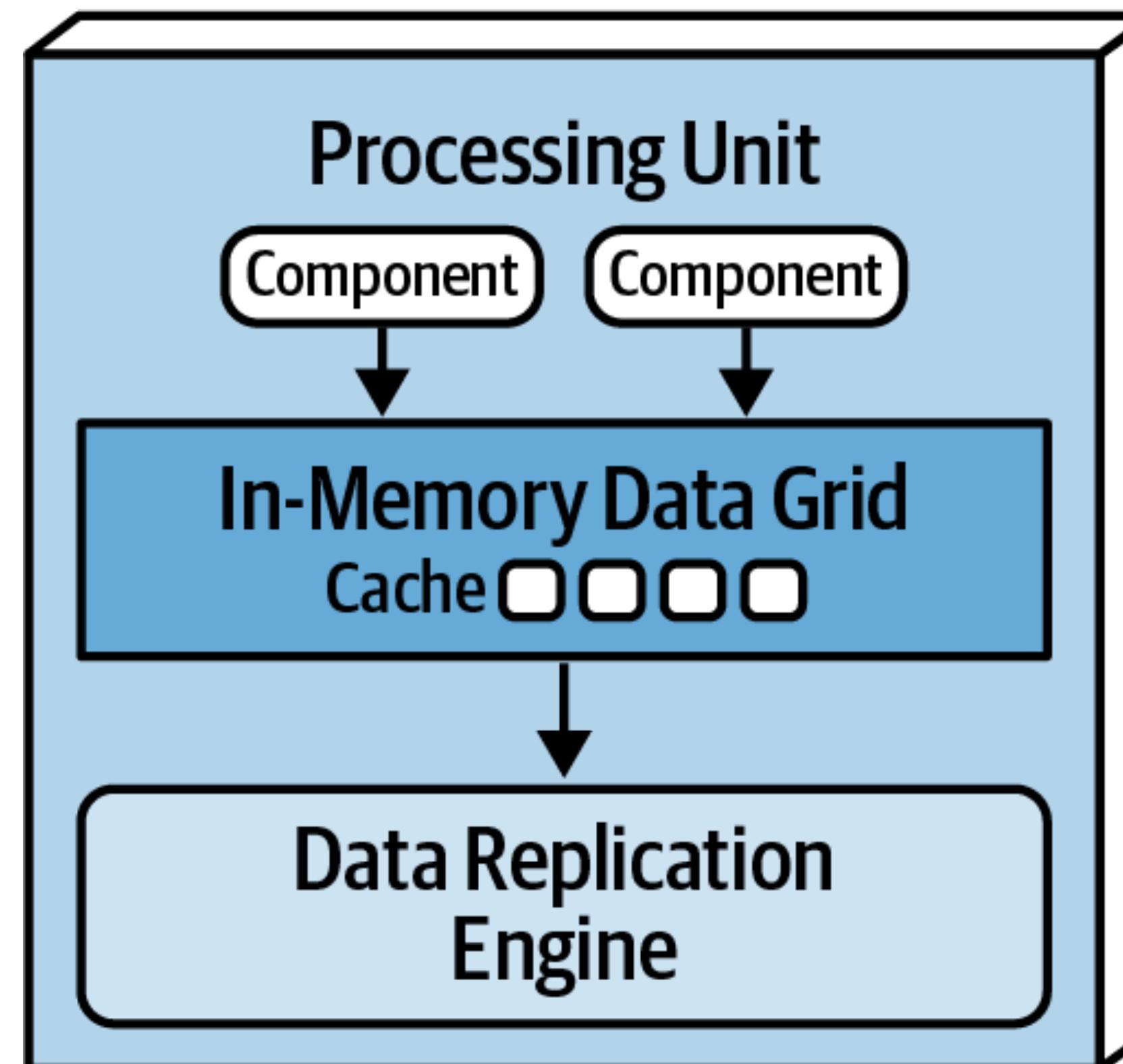
# space-based architecture



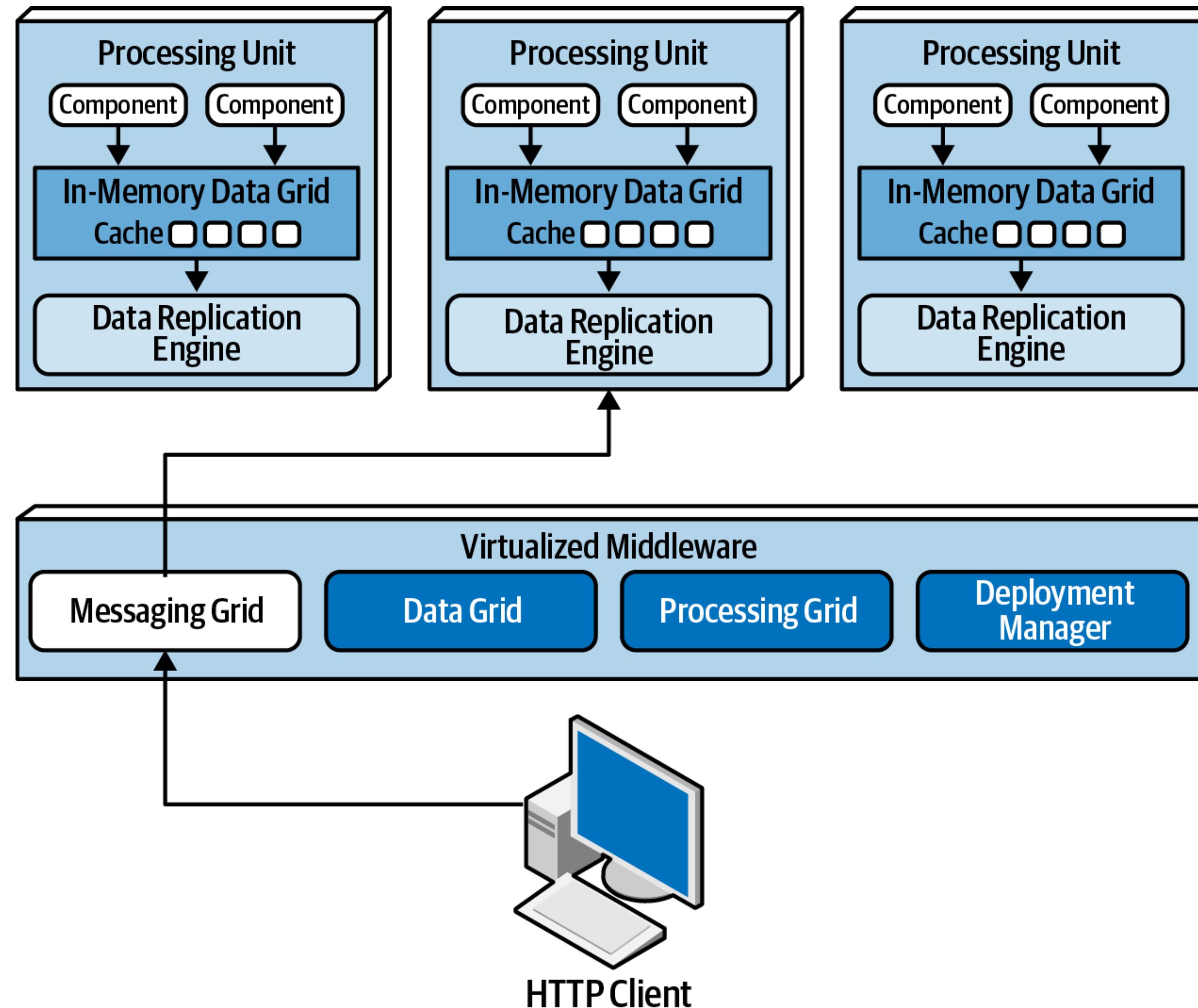
# space-based architecture



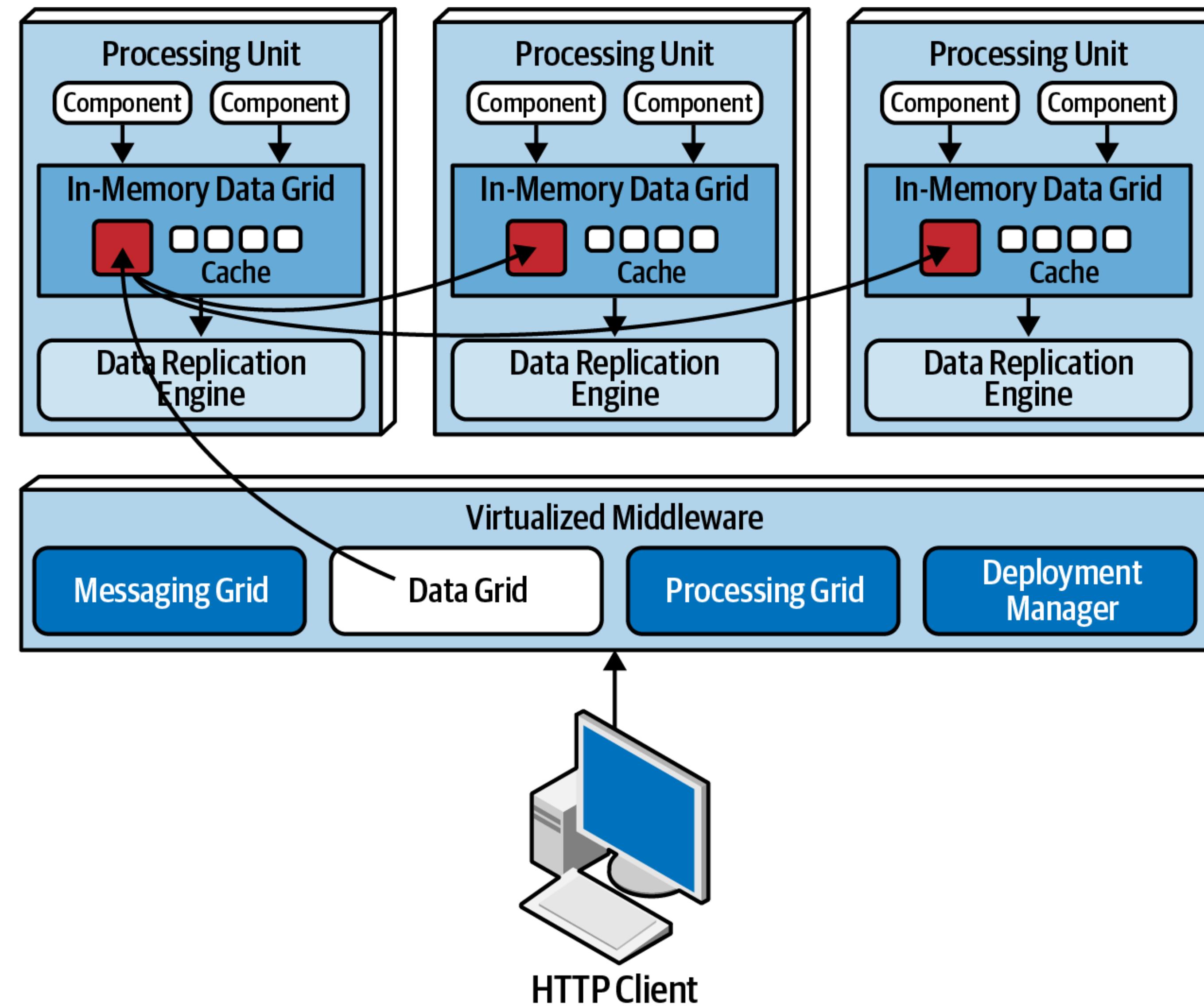
# space-based architecture



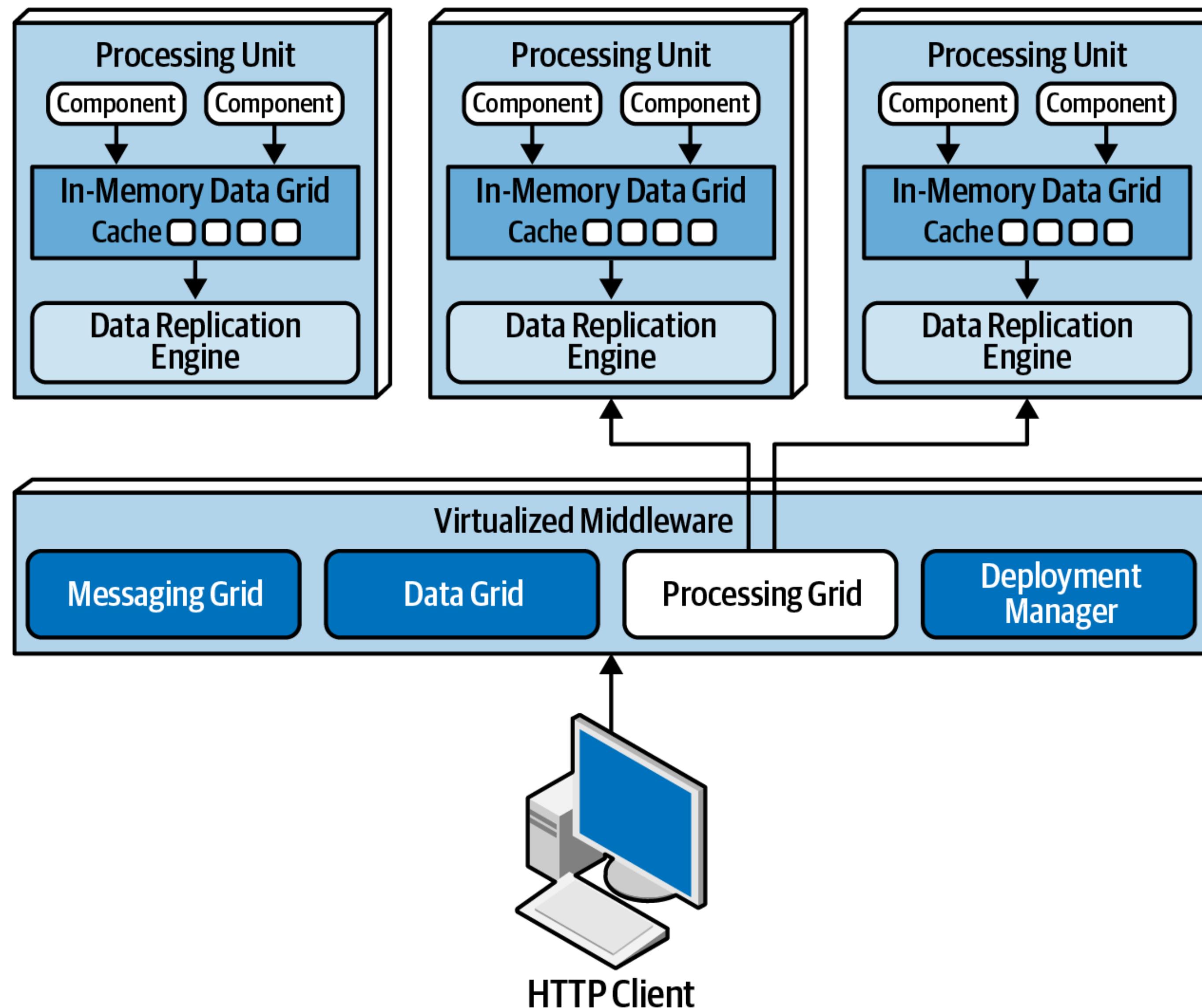
# space-based architecture



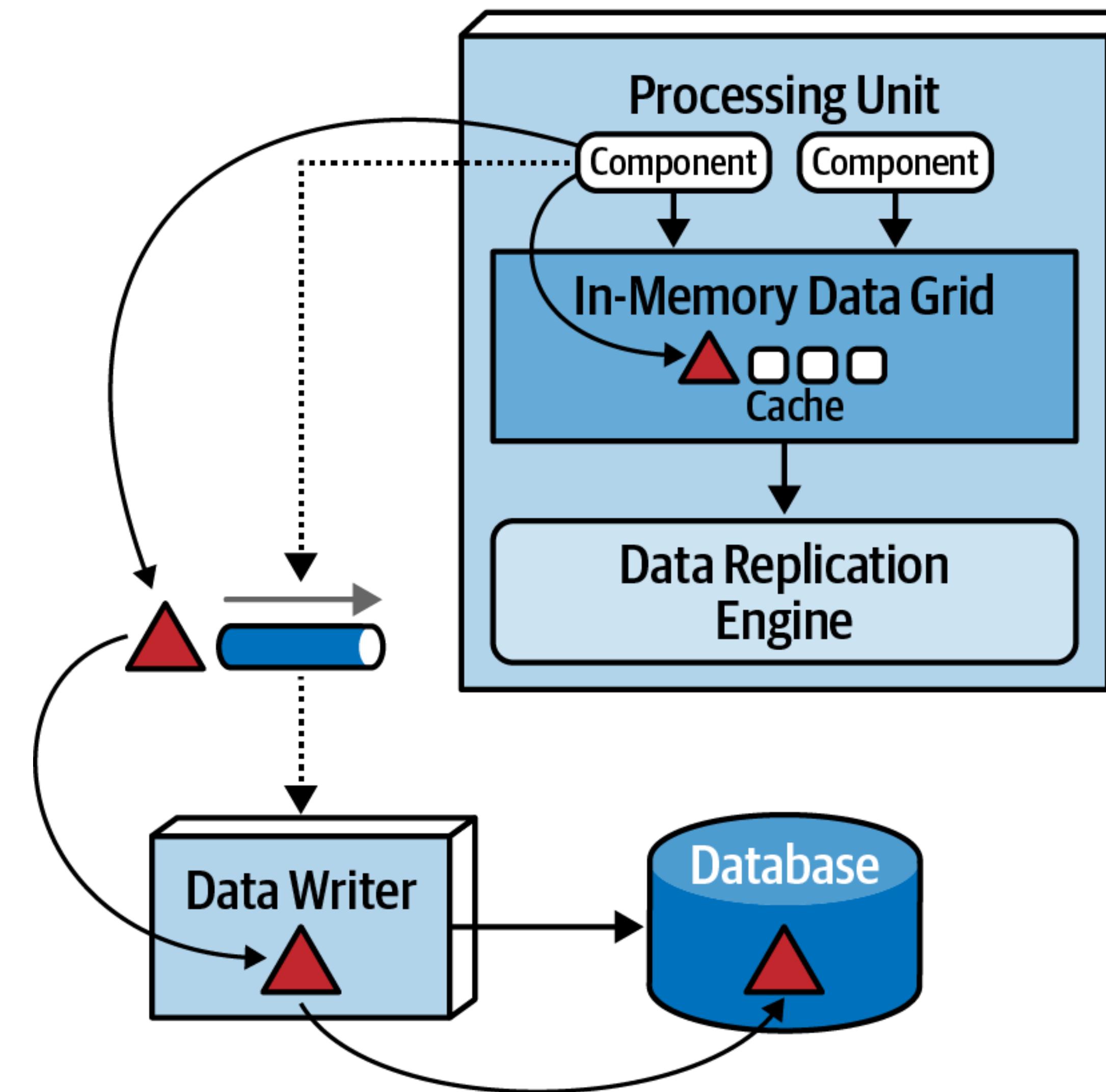
# space-based architecture



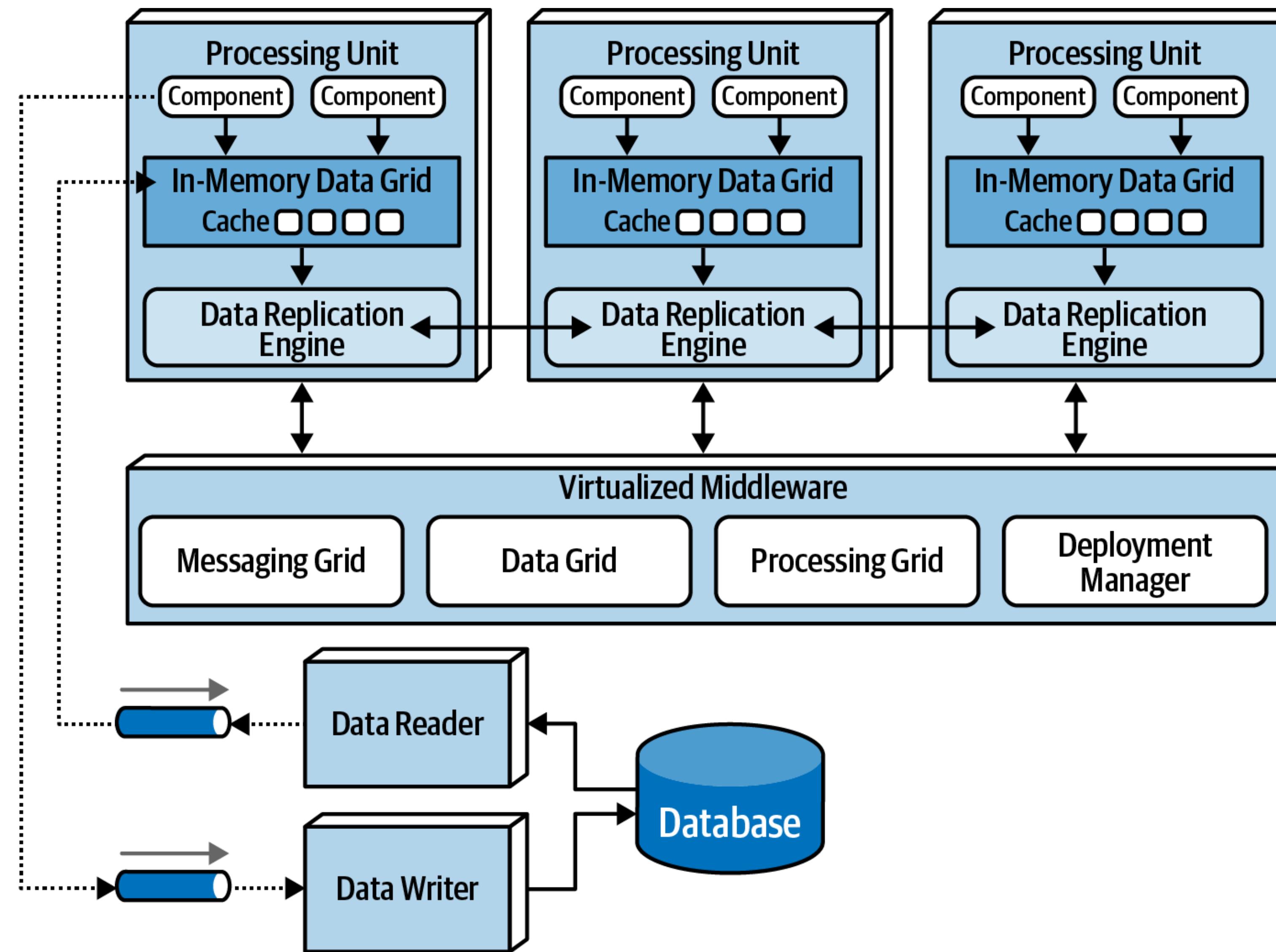
# space-based architecture



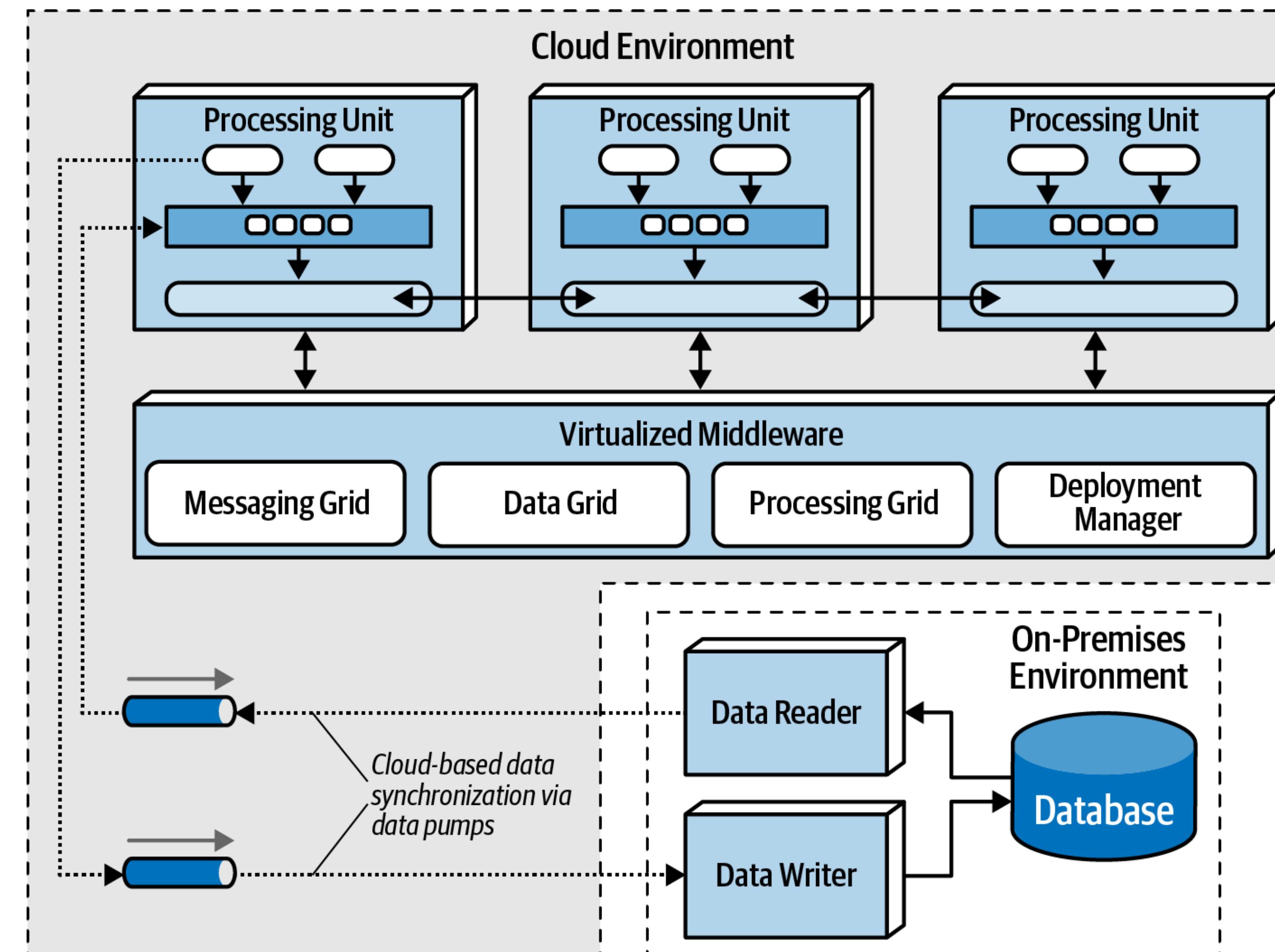
# space-based architecture



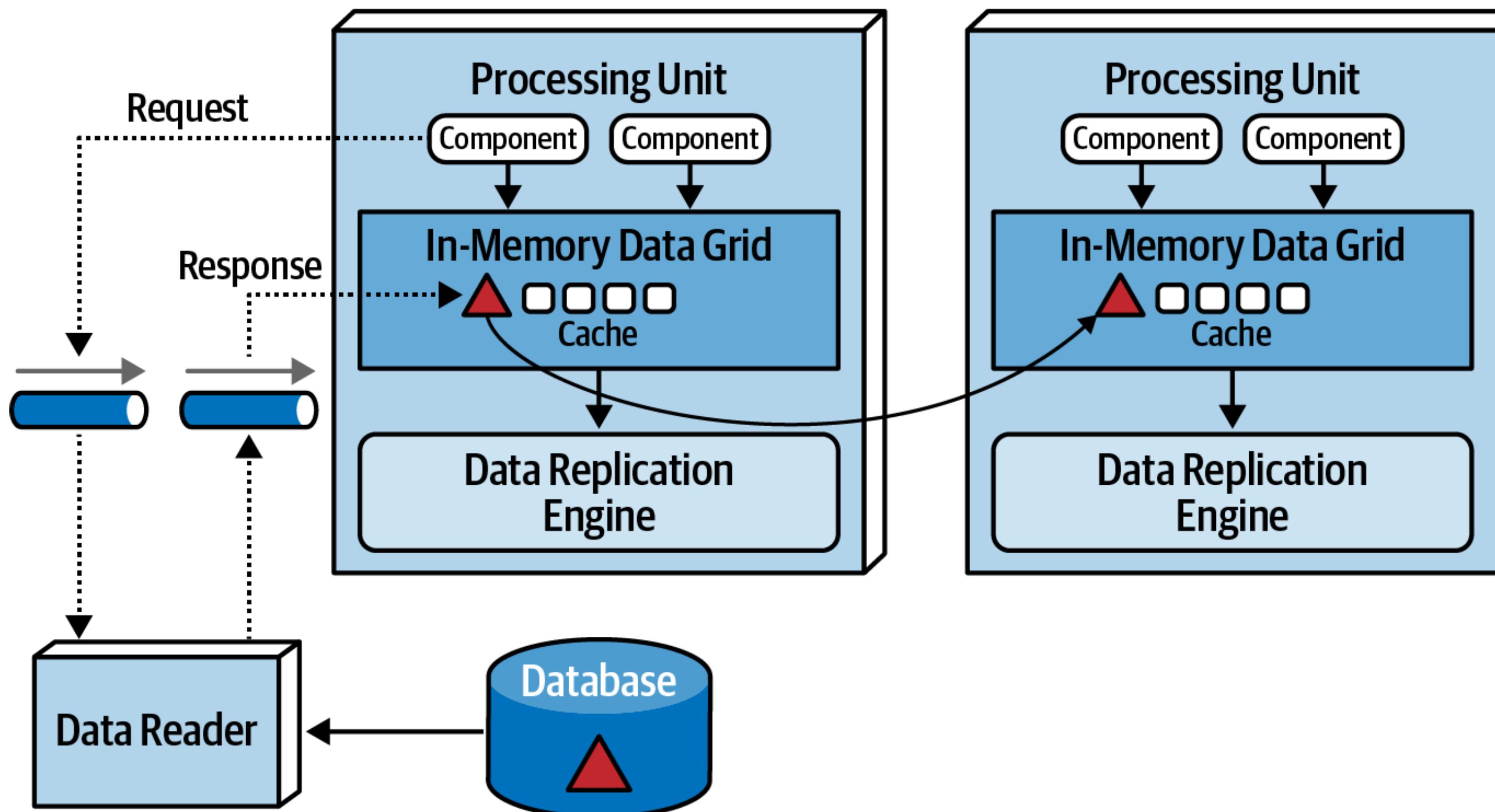
# space-based architecture



# space-based architecture



# suitability: space-based ?



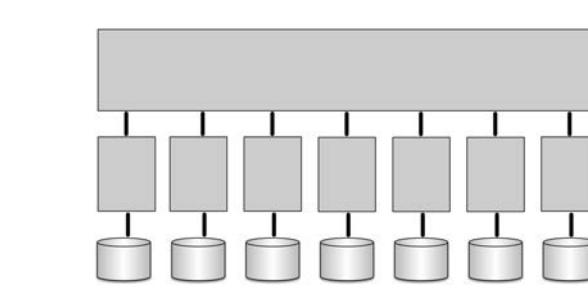
# Going Going Gone!



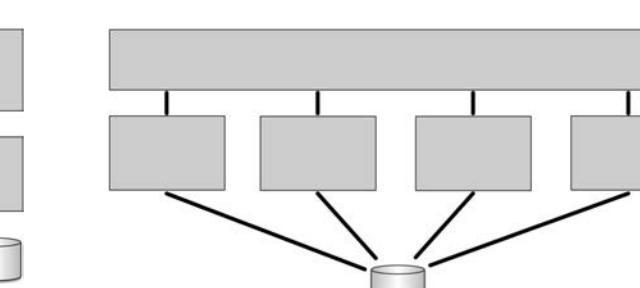
layered monolith



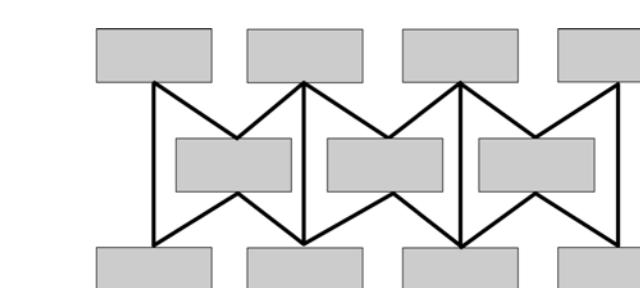
microkernel



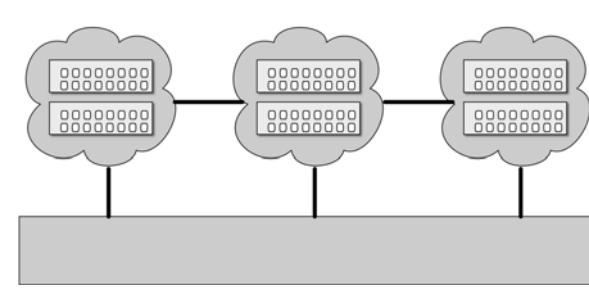
microservices



service-based



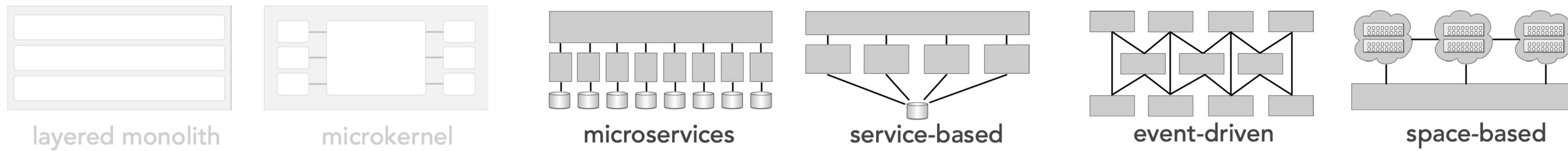
event-driven



space-based

	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★	★★★★★	★★★★★	★★★★	★★★
testability	★★	★★★	★★★★★	★★★★★	★★★	★
performance	★★★★★	★★★★★	★★	★★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★★★	★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★
evolvability	★	★★★	★★★★★	★★★★★	★★★★★	★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★

# Going Going Gone!



	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
testability	★★	★★★	★★★★★	★★★★★	★★★	★
performance	★★★	★★★	★	★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★★	★★★★★	★★★★★
elasticity	★	★	★★★	★★★	★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★★
evolvability	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★

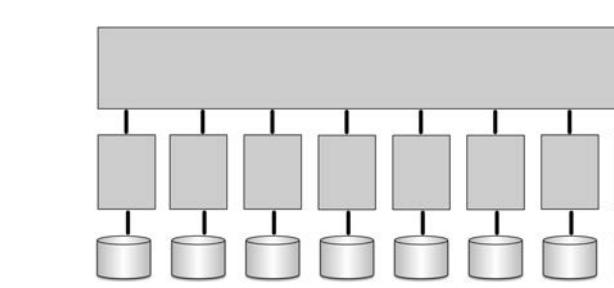
# Going Going Gone!



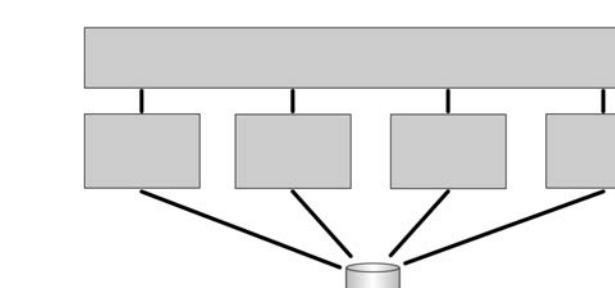
layered monolith



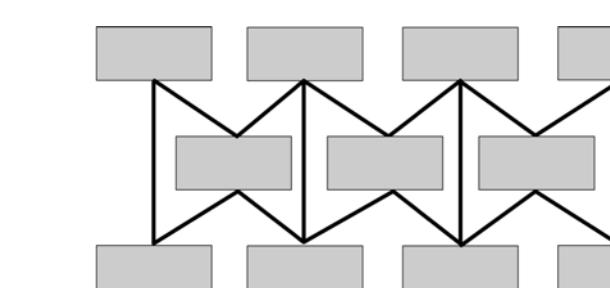
microkernel



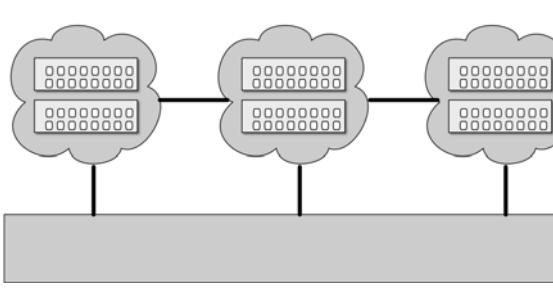
microservices



service-based



event-driven



space-based

	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
testability	★★	★★★	★★★★★	★★★★★	★★★	★★
performance	★★★★★	★★★★★	★	★★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★★★★★	★★★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★★★	★	★★★★★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★★
evolvability	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★★

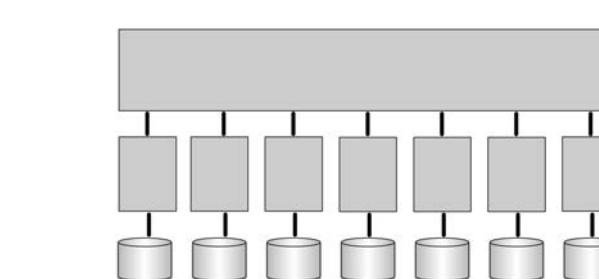
# Going Going Gone!



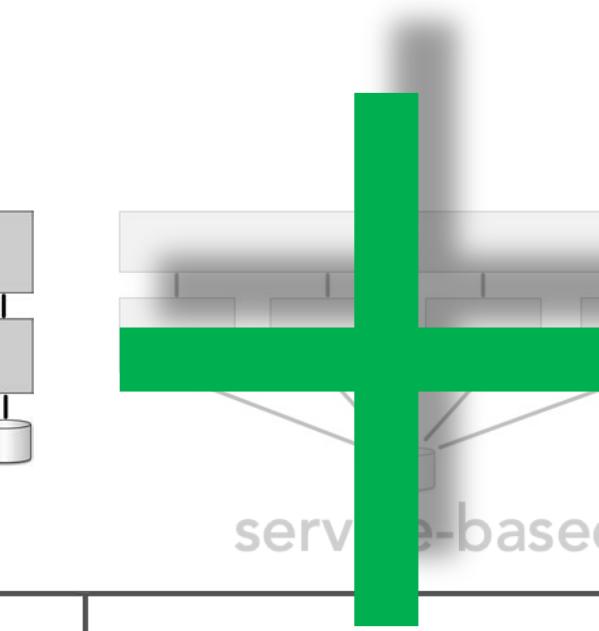
layered monolith



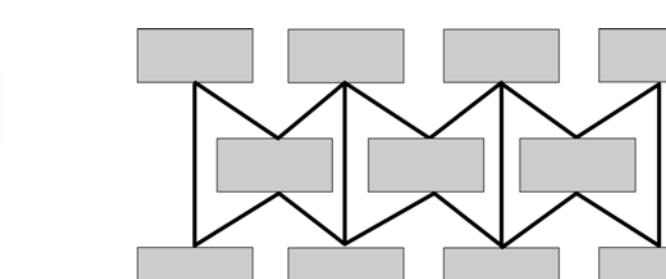
microkernel



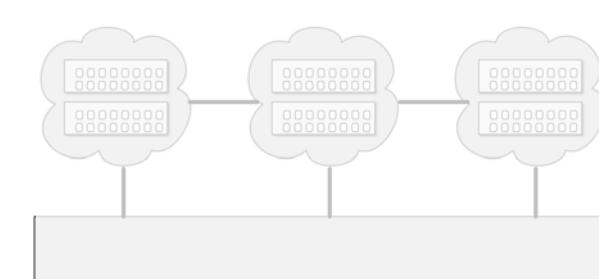
microservices



service-based



event-driven

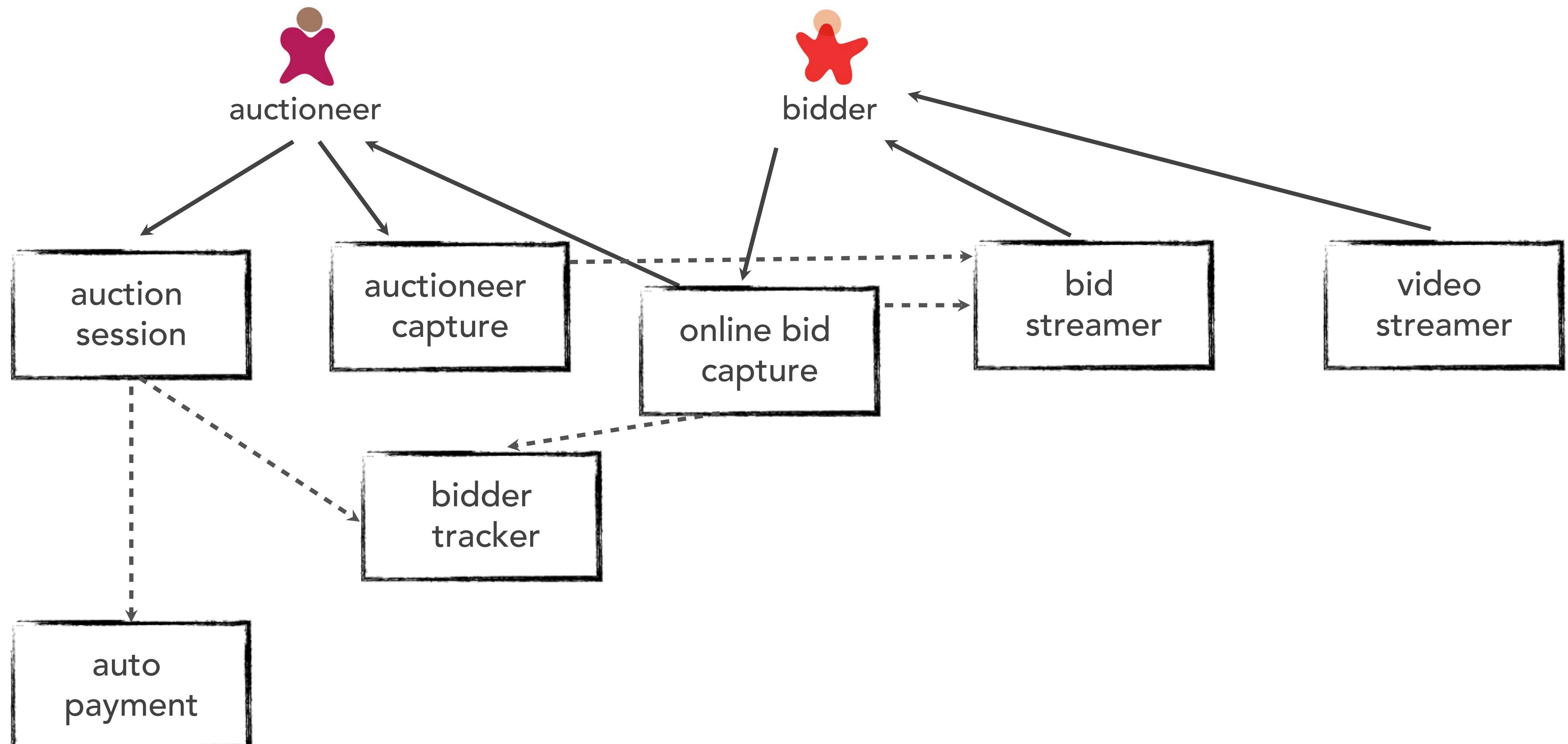


space-based

	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
deployment	★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
testability	★★	★★★★★	★★★★★	★★★★★	★★★	★
performance	★★★★★	★★★★★	★★★	★★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★★★	★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★★★
evolvability	★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★★

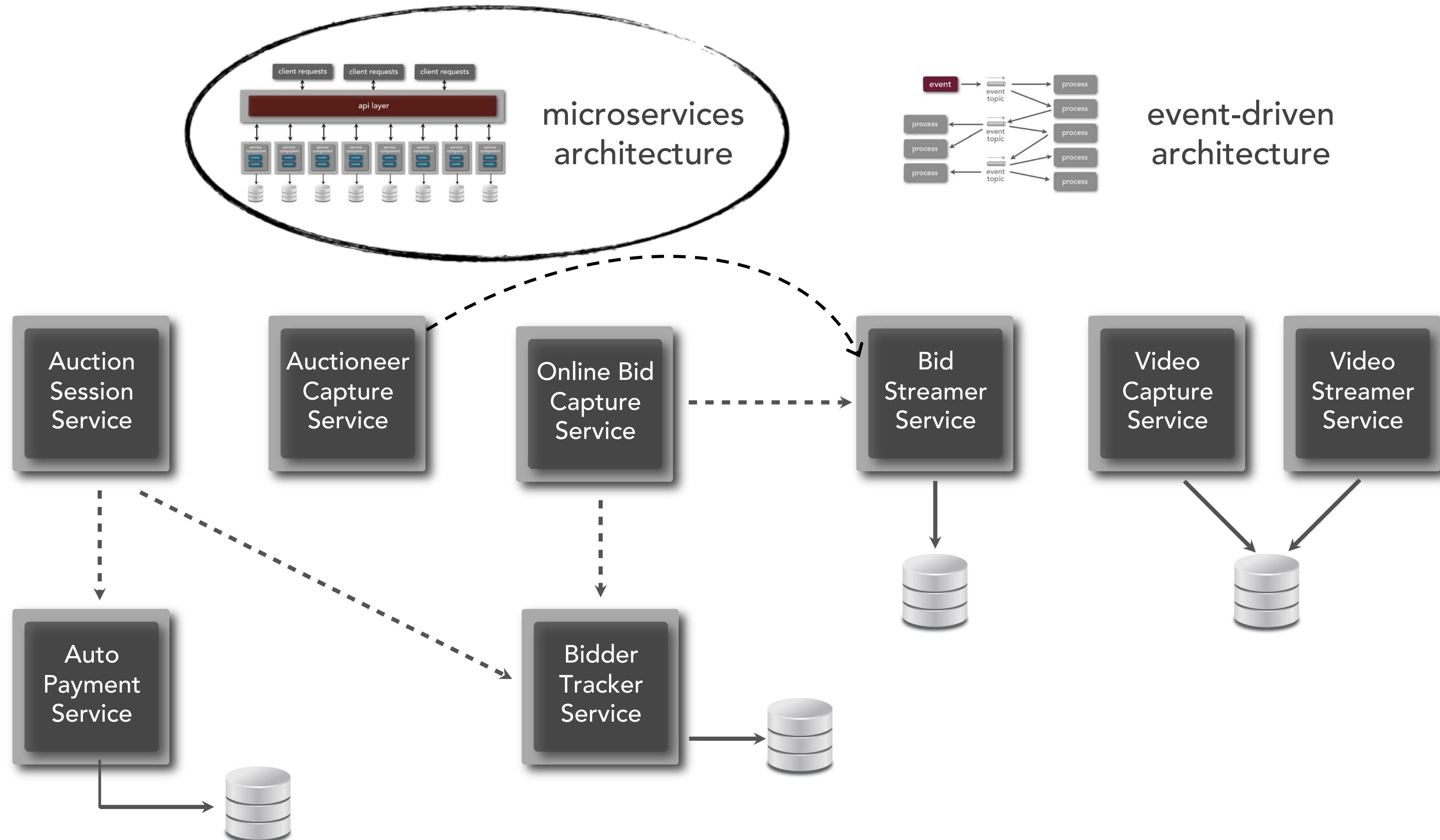
Your Architectural Kata is...

# Going Going Gone!



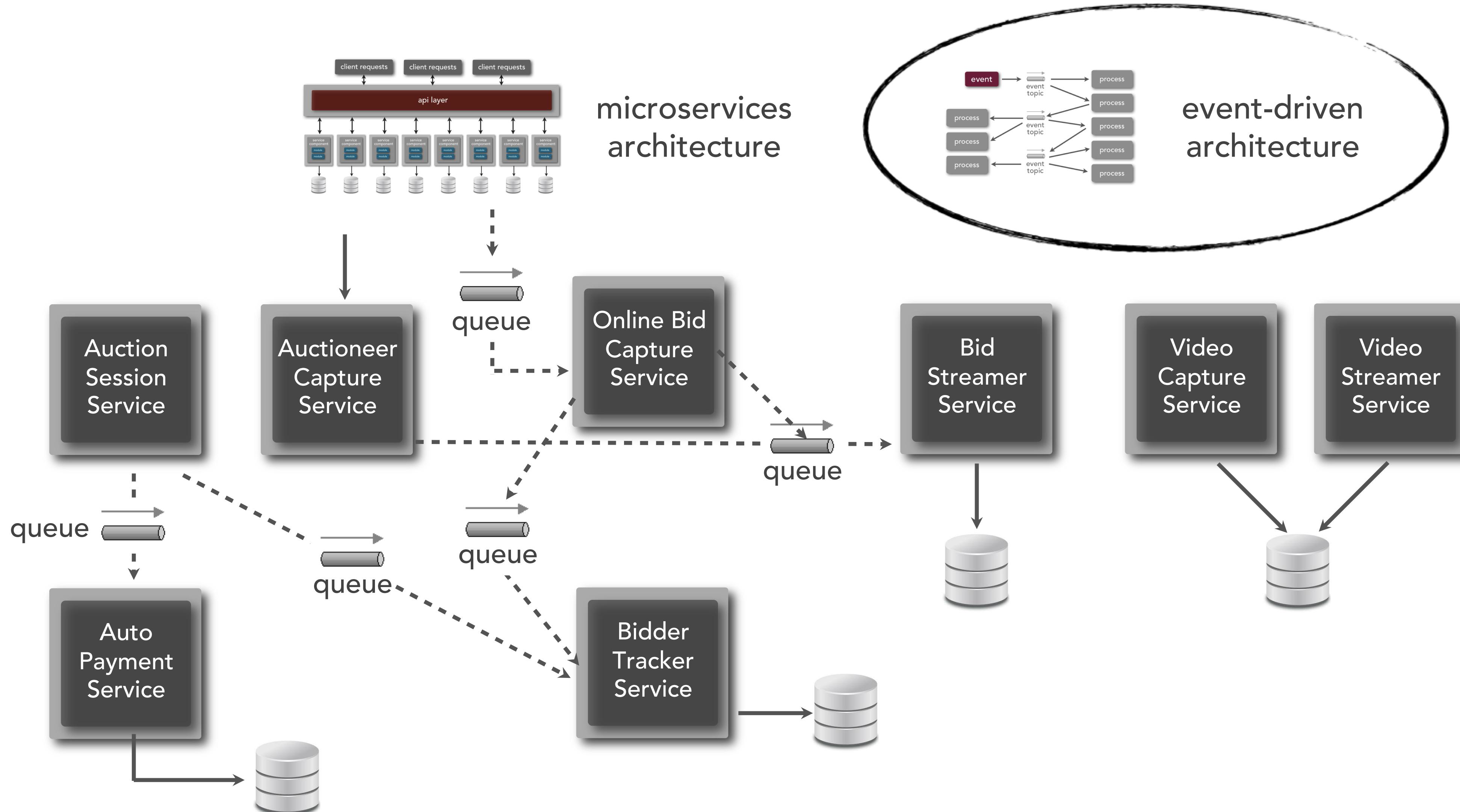
Your Architectural Kata is...

# Going Going Gone!



Your Architectural Kata is...

# Going Going Gone!



Your Architectural Kata is...

# Silicon Sandwiches

A national sandwich shop wants to enable 'fax in your order' but over the Internet instead (in addition to their current fax-in service)

- **Users:** thousands, perhaps one day millions
- **Requirements:**
  - users will place their order, then be given a time to pick up their sandwich and directions to the shop (which must integrate with several external mapping services that include traffic information)
  - if the shop offers a delivery service, dispatch the driver with the sandwich to the user
  - mobile-device accessibility
  - offer national daily promotions/specials
  - offer local daily promotions/specials
  - accept payment online or in person/on delivery

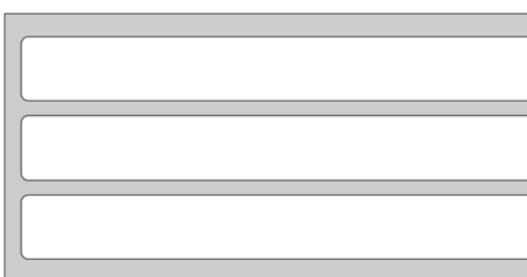
availability reliability

scalability
- **Additional Context:**
  - Sandwich shops are franchised, each with a different owner.
  - Parent company has near-future plans to expand overseas.
  - Corporate goal is to hire inexpensive labor to maximize profit.
  - Time to market is critical.

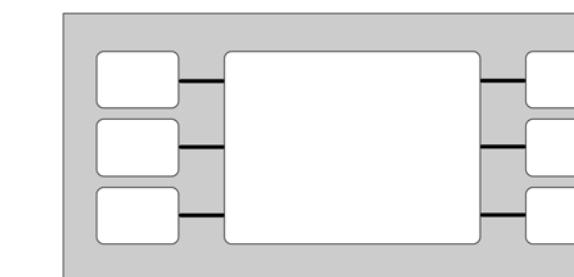
Customizability

{ location  
sales  
recipe

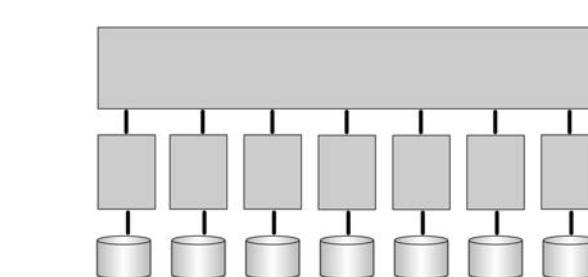
# Silicon Sandwiches



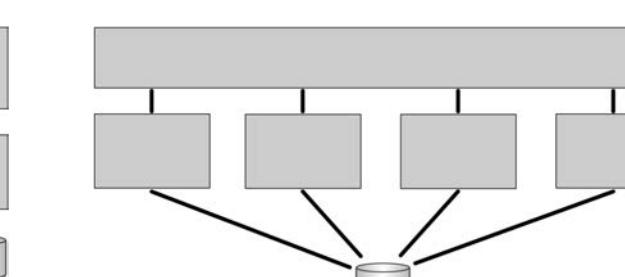
layered monolith



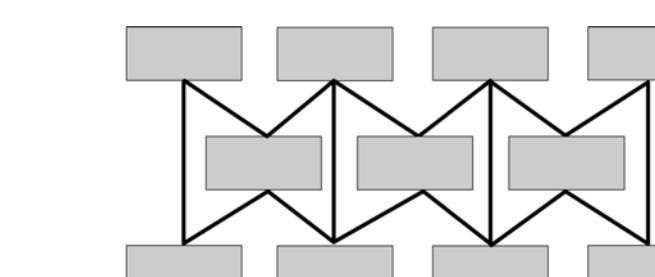
microkernel



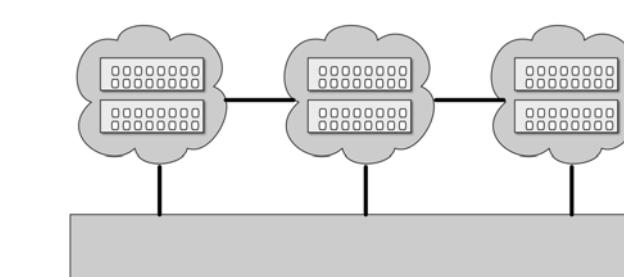
microservices



service-based



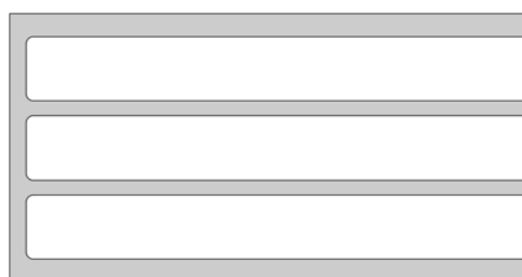
event-driven



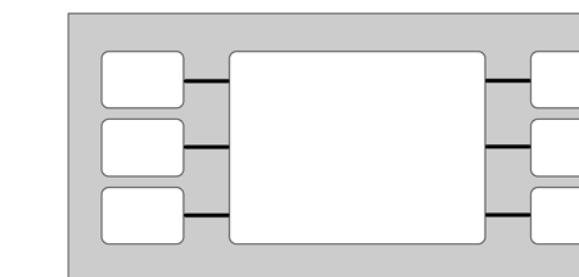
space-based

	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★★	★★★★★	★★★★★	★★★★	★★★★
testability	★★	★★★★	★★★★★	★★★★	★★★	★
performance	★★★★★	★★★★★	★	★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★★★	★★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★
evolvability	★	★★★★	★★★★★	★★★★★	★★★★★	★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★

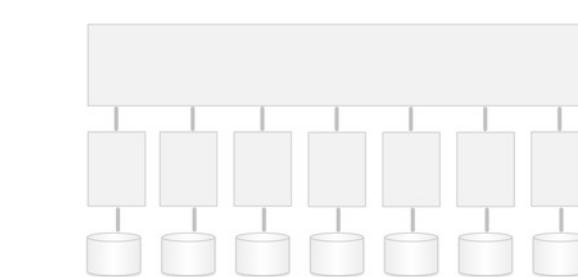
# Silicon Sandwiches



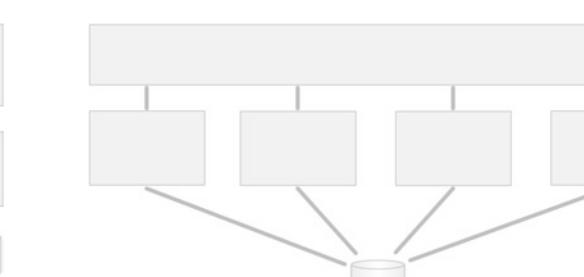
layered monolith



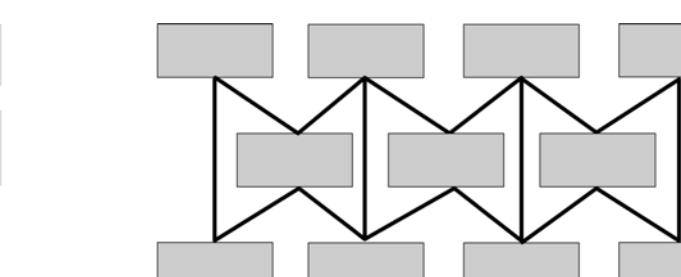
microkernel



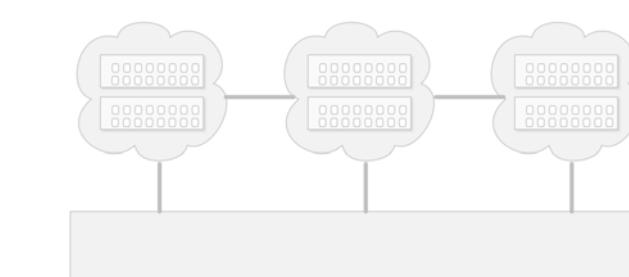
microservices



service-based



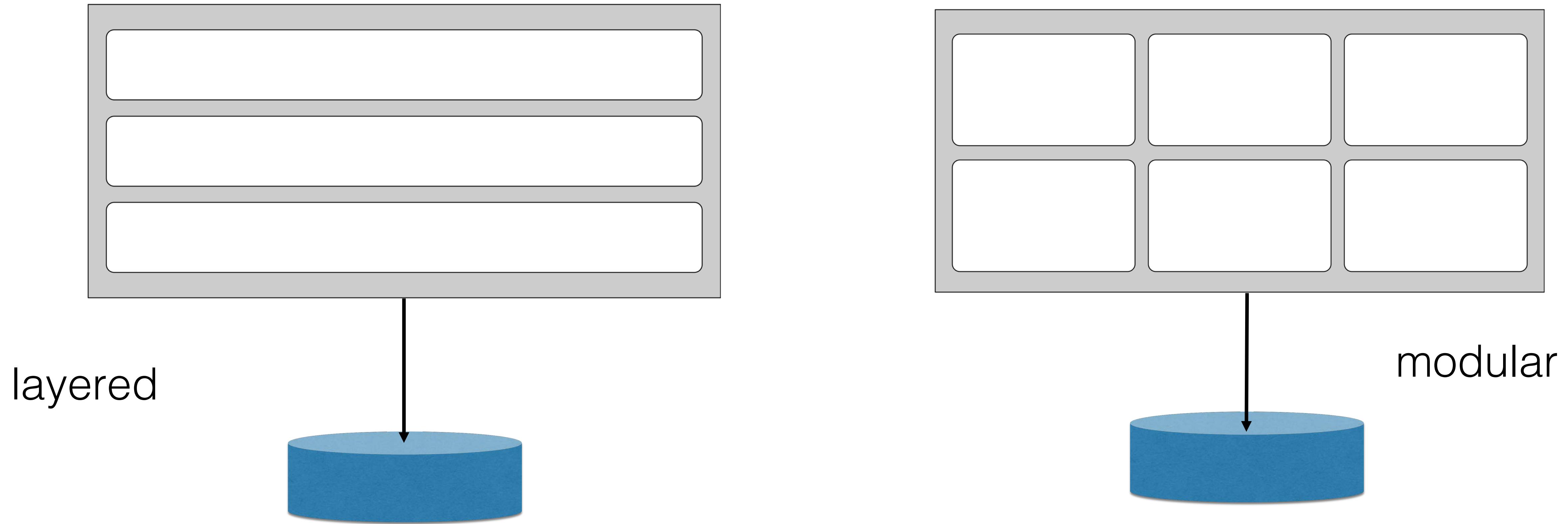
event-driven



space-based

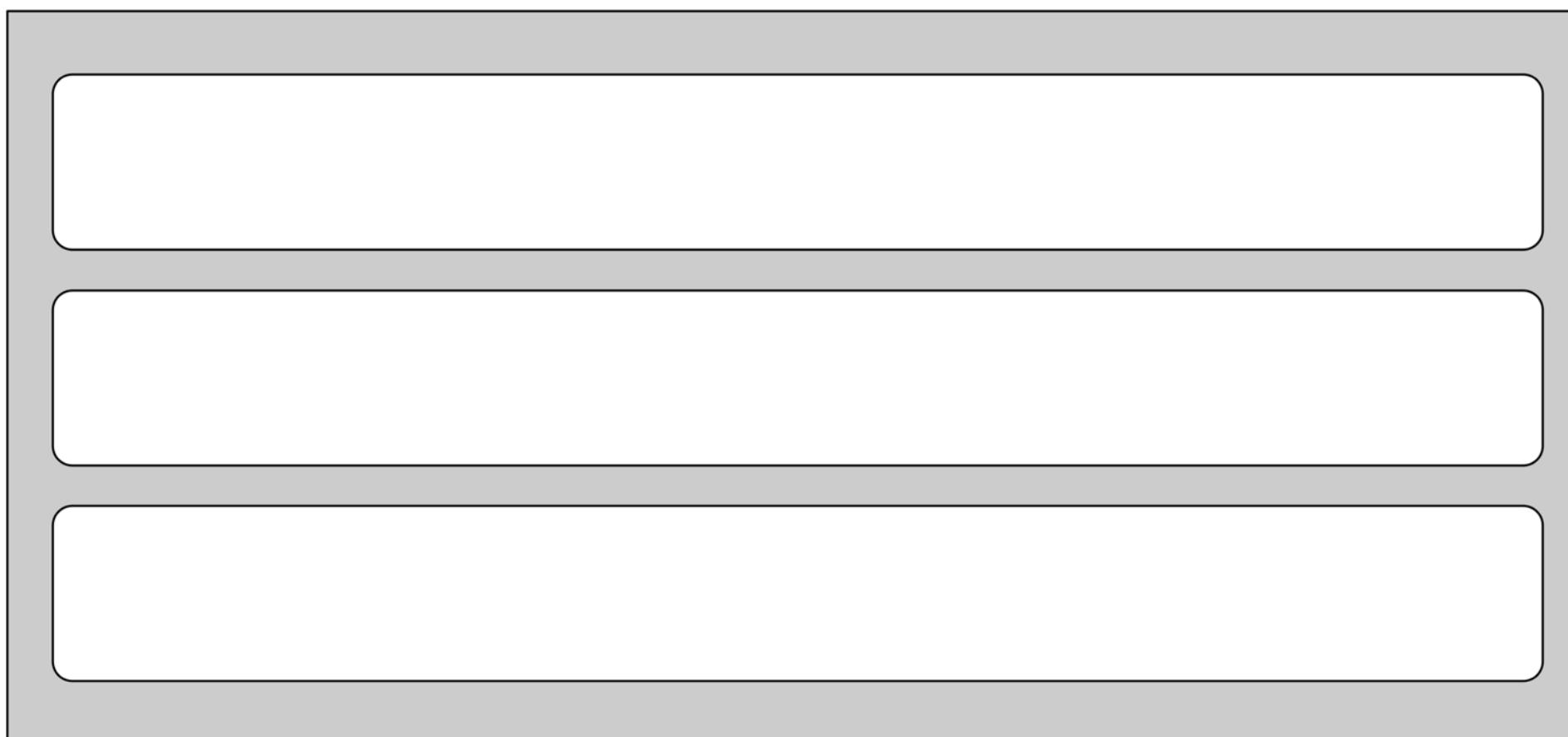
	layered monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★★	★★★★	★★★★	★★★★	★★★★
testability	★★★	★★★★	★★★★	★★★	★★★	★
performance	★★★★★	★★★★★	★	★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★	★★★★	★★★★★
simplicity	★★★★★★	★★★★★★	★	★★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★★
evolvability	★	★★★★	★★★★★	★★★★★	★★★★★	★★★★
total cost	★★★★★★	★★★★★★	★	★★★★★	★★★	★★★

# monoliths



# monoliths

technical partitioning



layered

domain partitioning



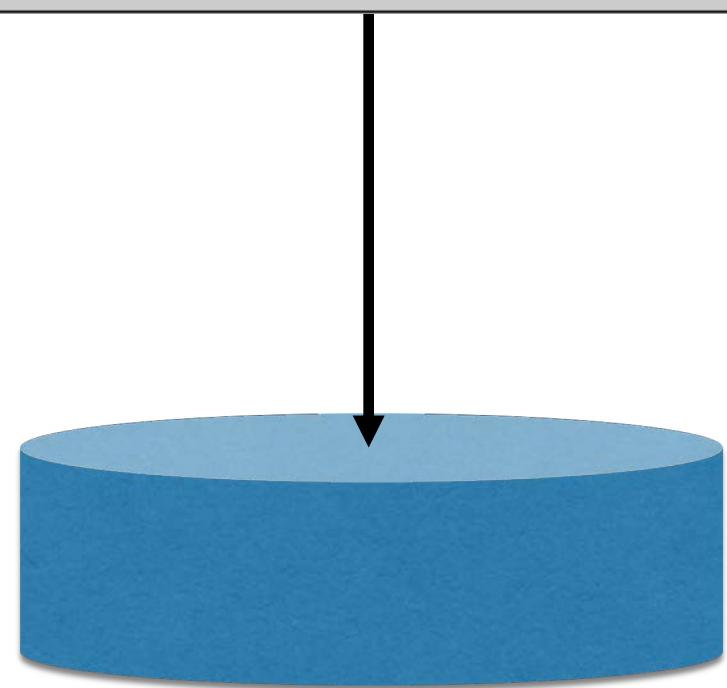
modular

# monoliths

technical partitioning

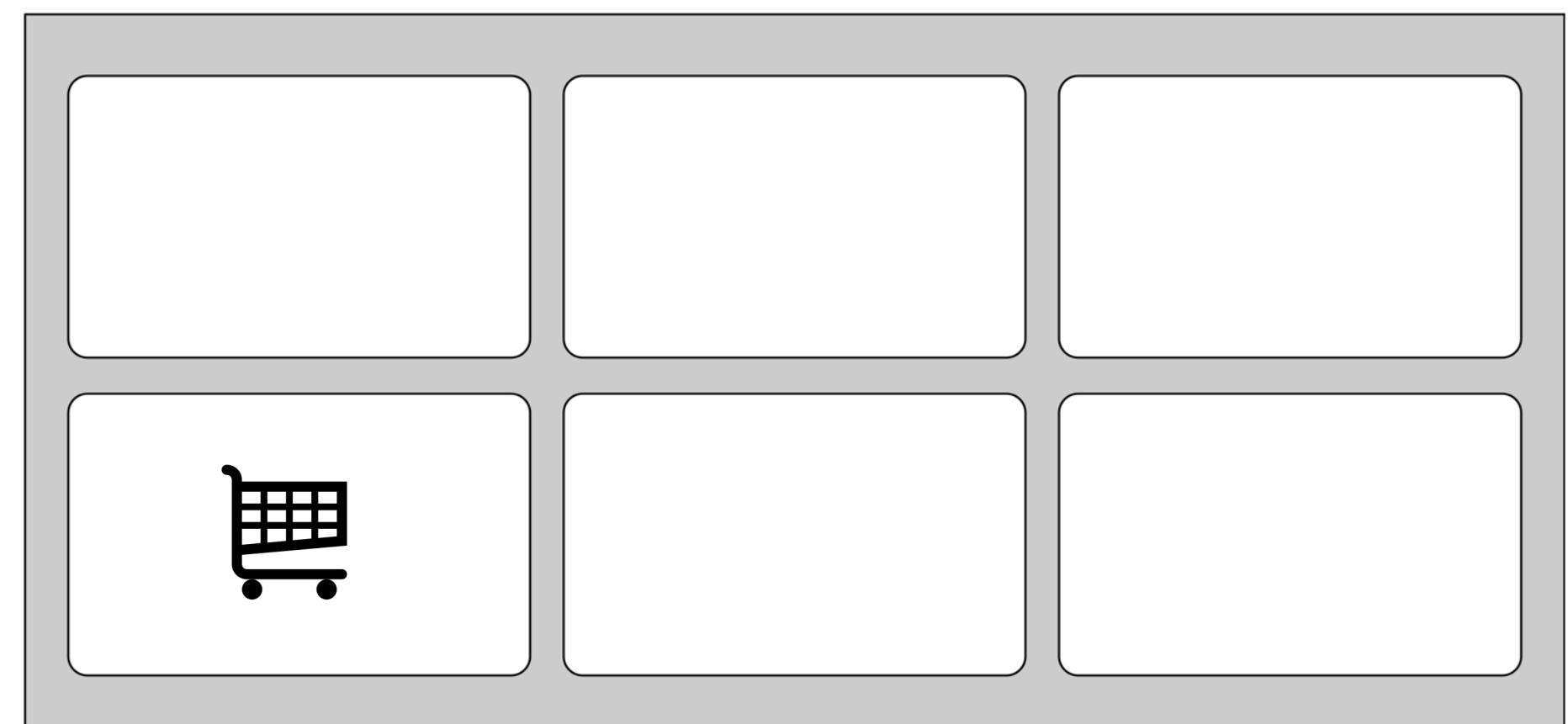


layered

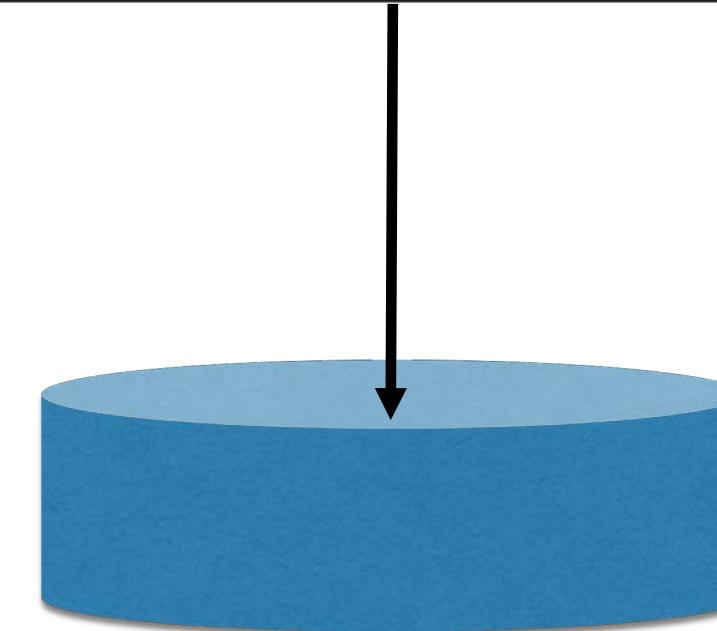


catalog checkout?

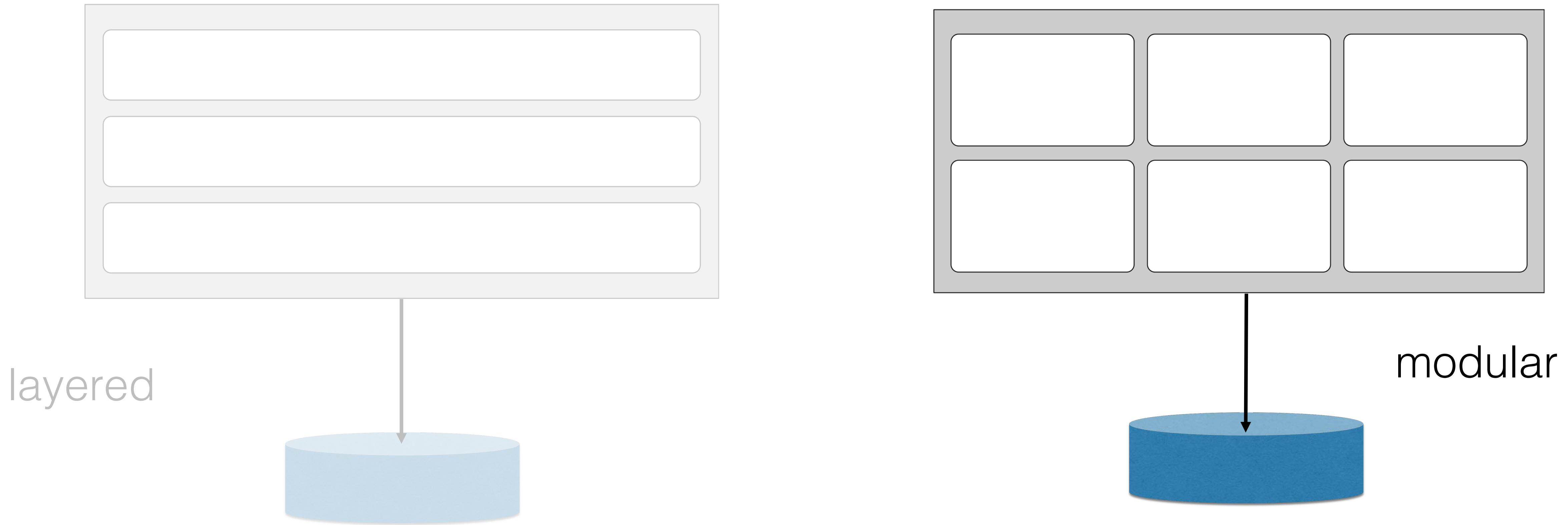
domain partitioning



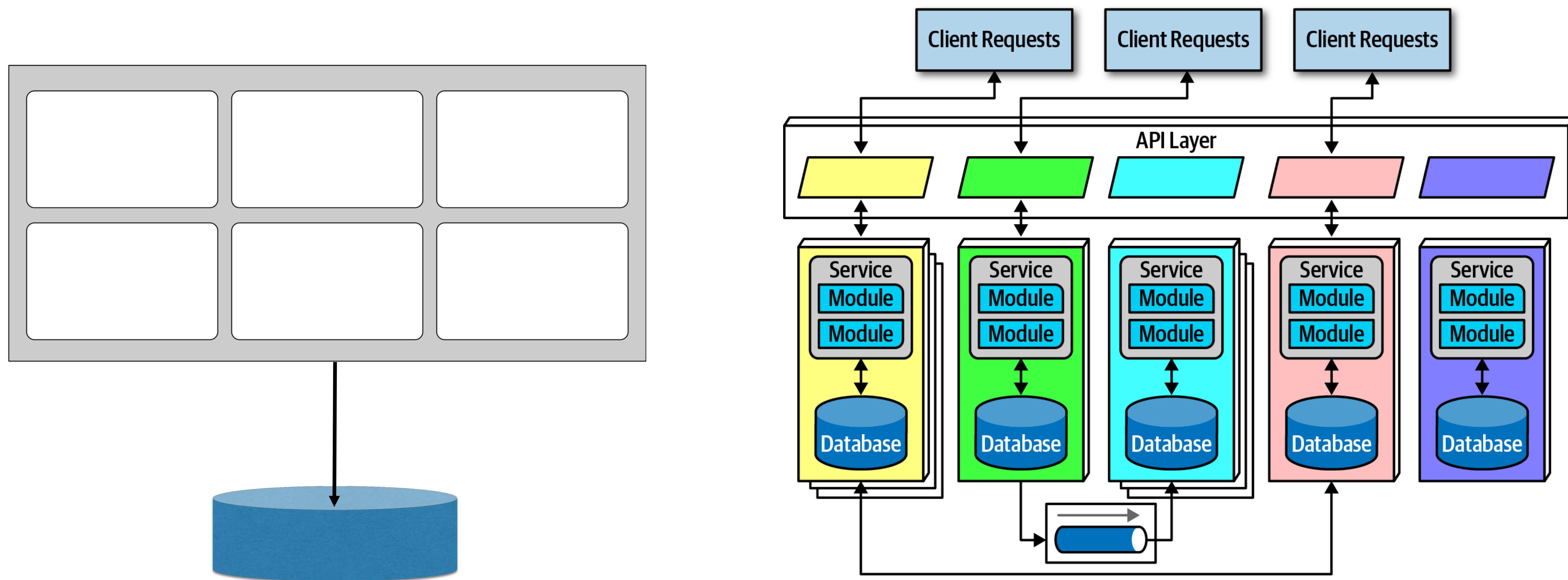
modular



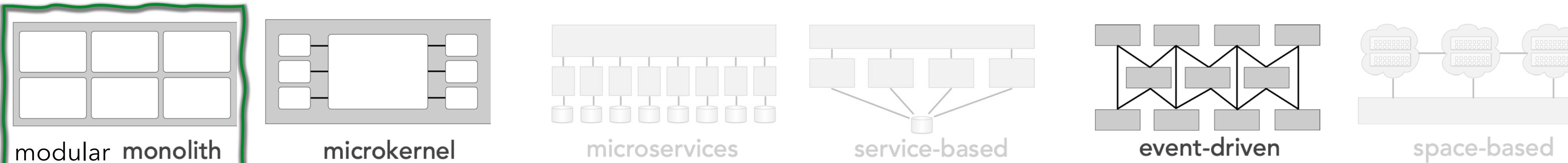
# monoliths



# modular monoliths → microservices



# Silicon Sandwiches

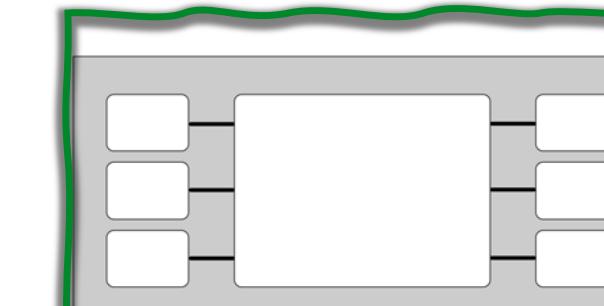


	modular monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
deployment	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
testability	★★	★★★	★★★★★	★★★★★	★★★	★
performance	★★★★★	★★★★★	★	★★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★	★★★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★★
evolvability	★	★★★	★★★★★	★★★★★	★★★★★	★★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★★

# Silicon Sandwiches



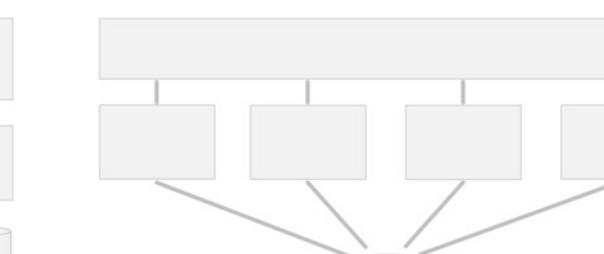
modular monolith



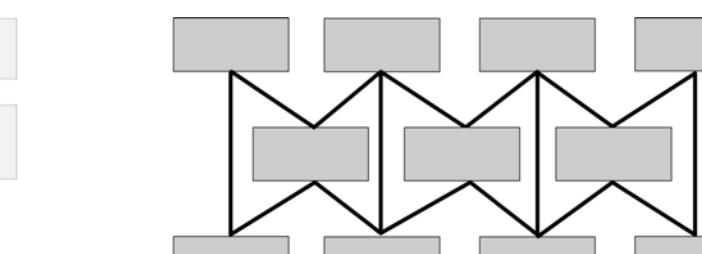
microkernel



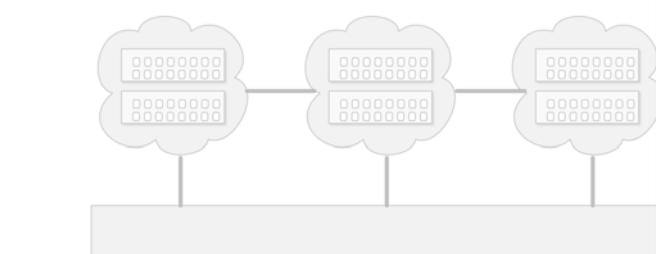
microservices



service-based



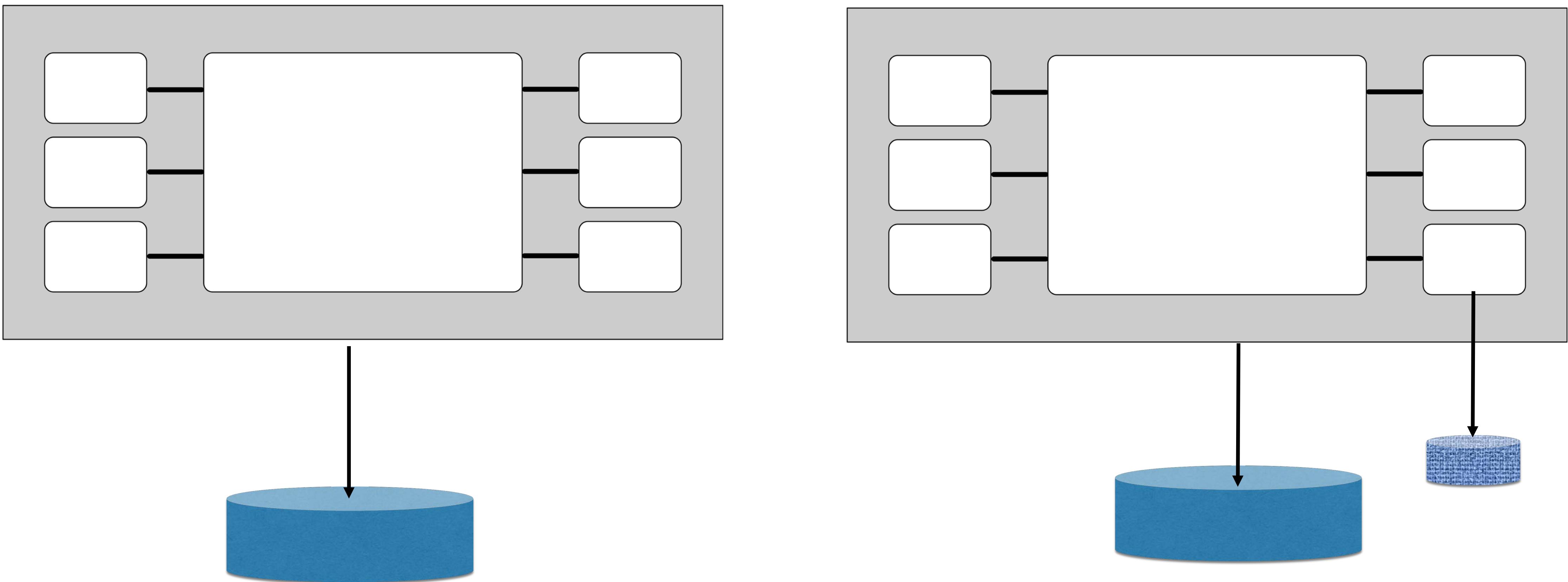
event-driven



space-based

	modular monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★	★★★	★★★	★★
deployment	★	★★★	★★★	★★★	★★★	★★★
testability	★★	★★★	★★★	★★★	★★	★
performance	★★★	★★★	★	★★★	★★★	★★★
scalability	★	★	★★★	★★★	★★★	★★★
elasticity	★	★	★★★	★	★★★	★★★
simplicity	★★★★★	★★★★★	★	★★★	★	★
fault-tolerance	★	★	★★★	★★★	★★★	★★★
evolvability	★	★★★	★★★	★★★	★★★	★★★
total cost	★★★★★	★★★★★	★	★★★	★★★	★★

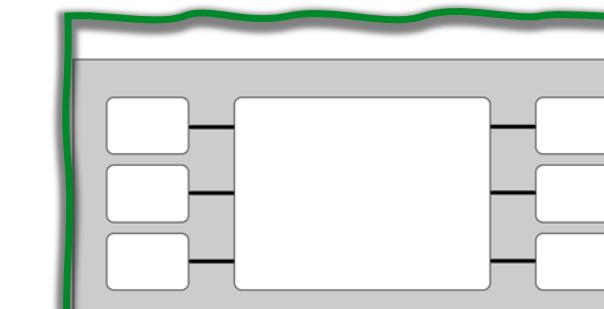
# microkernel



# Silicon Sandwiches



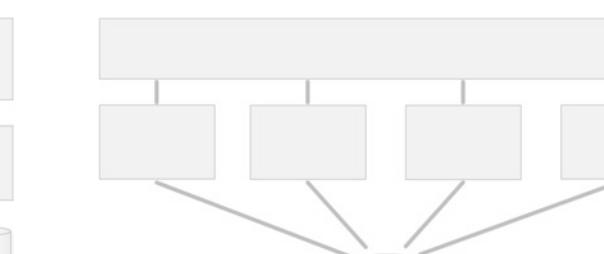
modular monolith



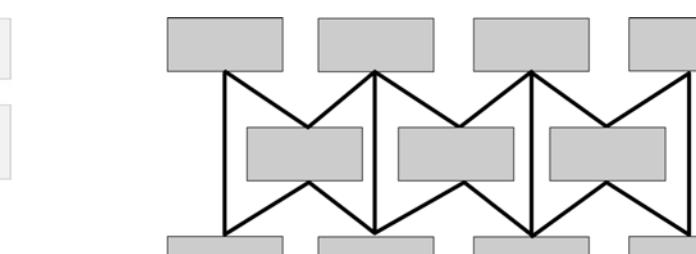
microkernel



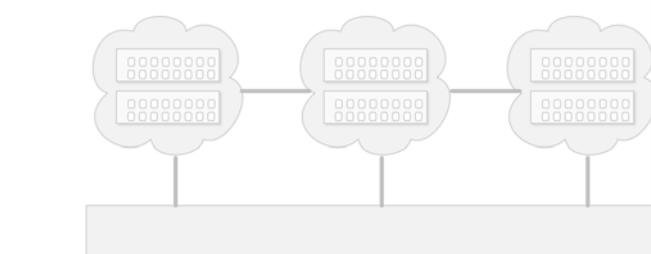
microservices



service-based



event-driven



space-based

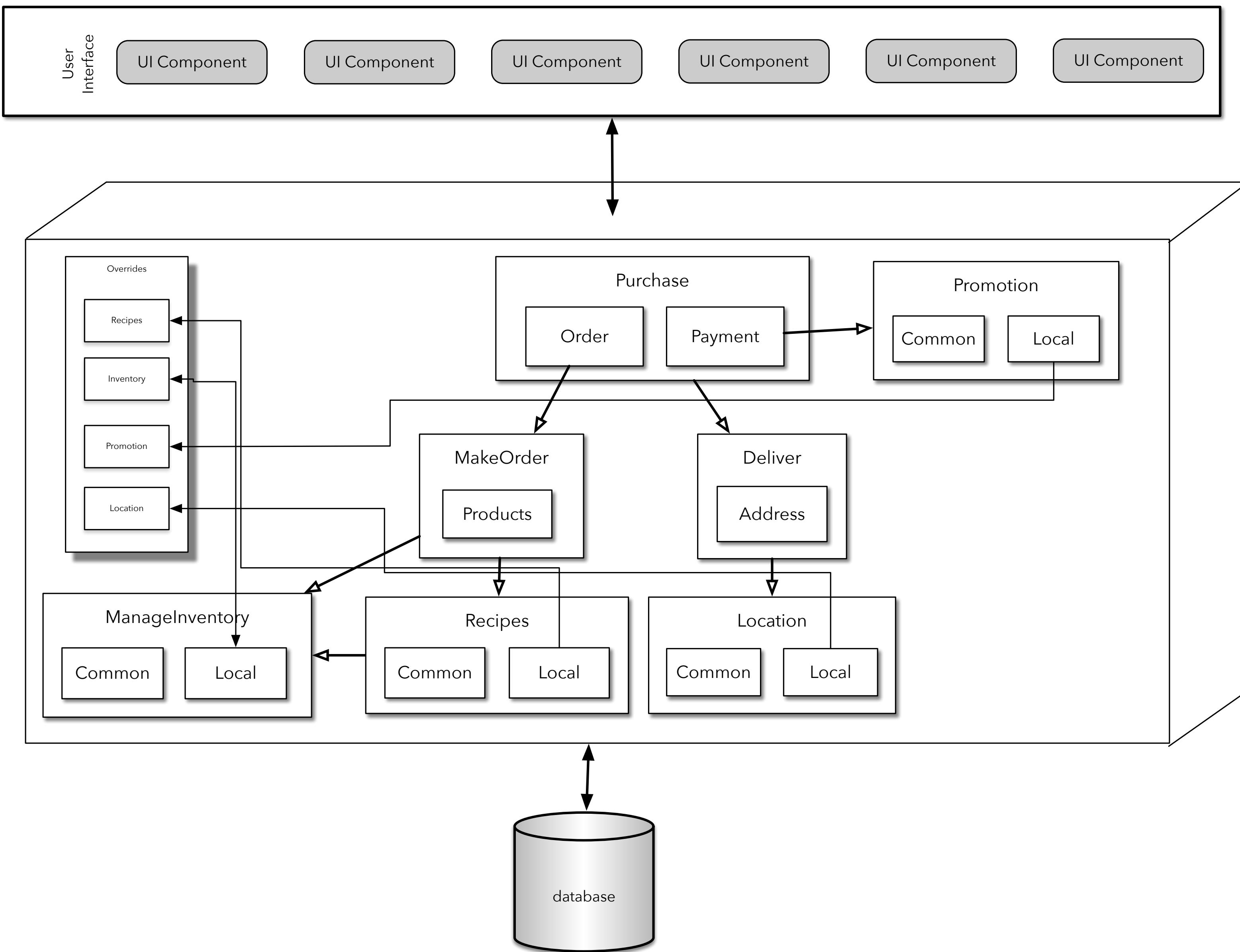
	modular monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★	★★★	★★★	★★★	★★
deployment	★	★★★	★★★	★★★	★★★	★★★
testability	★★	★★★	★★★	★★★	★★	★
performance	★★★	★★★	★	★★★	★★★	★★★
scalability	★	★	★★★	★★★	★★★	★★★
elasticity	★	★	★★★	★	★★★	★★★
simplicity	★★★★★	★★★★★	★	★★★	★	★
fault-tolerance	★	★	★★★	★★★	★★★	★★★
evolvability	★	★★★	★★★	★★★	★★★	★★★
total cost	★★★★★	★★★★★	★	★★★	★★★	★★

# Silicon Sandwiches

	modular monolith	microkernel	microservices	service-based	event-driven	space-based
agility	★	★★★★★	★★★★★	★★★★★	★★★★★	★★★★★
deployment	★	★★★★	★★★★	★★★★	★★★★	★★★★
testability	★★	★★★★	★★★★	★★★★	★★★	★
performance	★★★★★	★★★★★	★	★★★★	★★★★★	★★★★★
scalability	★	★	★★★★★	★★★★	★★★★★	★★★★★
elasticity	★	★	★★★★★	★	★★★★★	★★★★★
simplicity	★★★★★	★★★★★	★	★★★★	★	★
fault-tolerance	★	★	★★★★★	★★★★★	★★★★★	★★★★
evolvability	★	★★★★	★★★★★	★★★★★	★★★★★	★★★★
total cost	★★★★★	★★★★★	★	★★★★★	★★★	★★

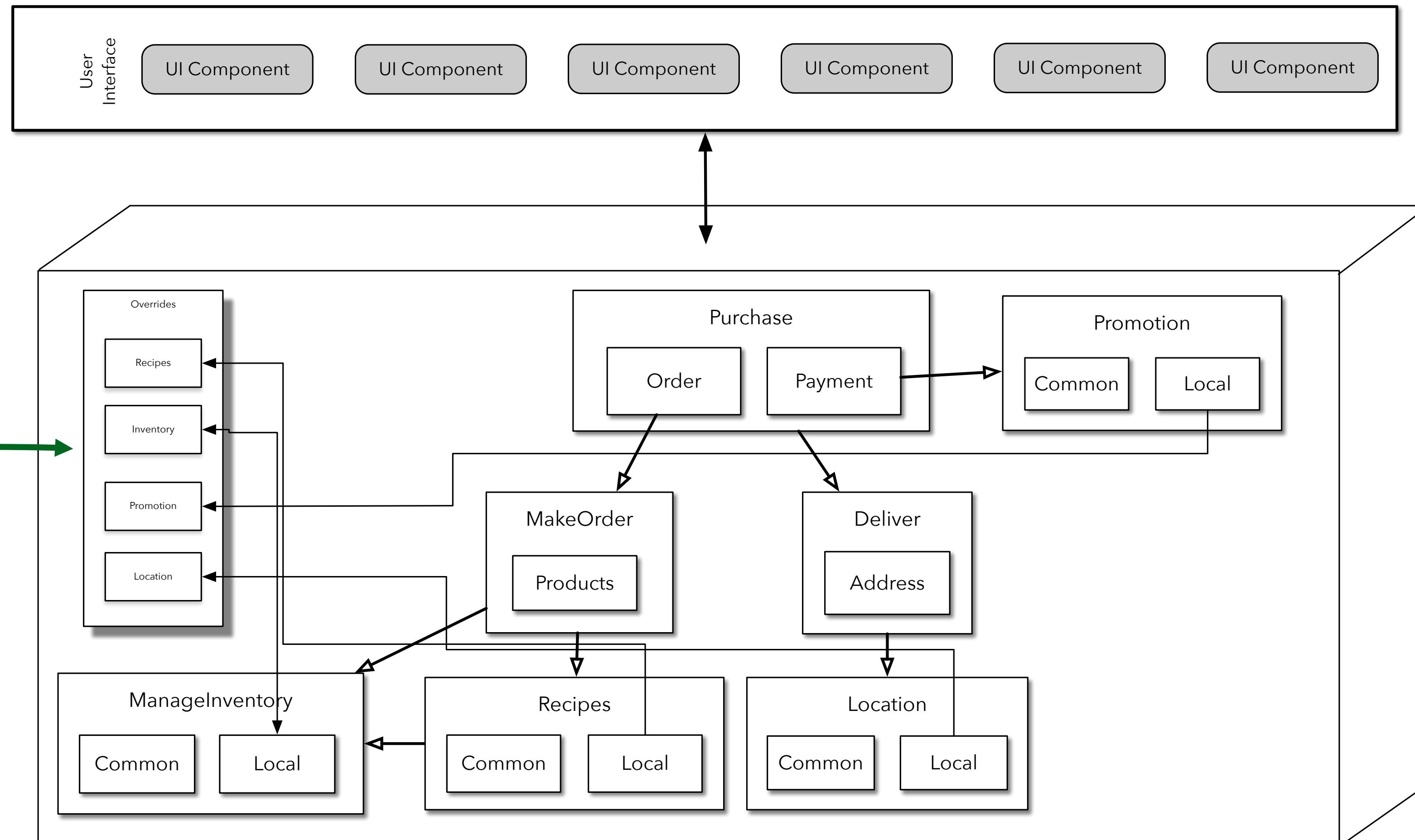
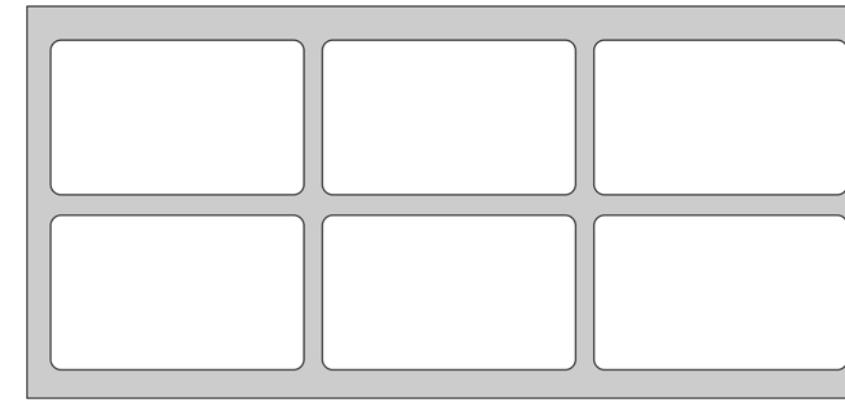
Your Architectural Kata is...

# Silicon Sandwiches



Your Architectural Kata is...

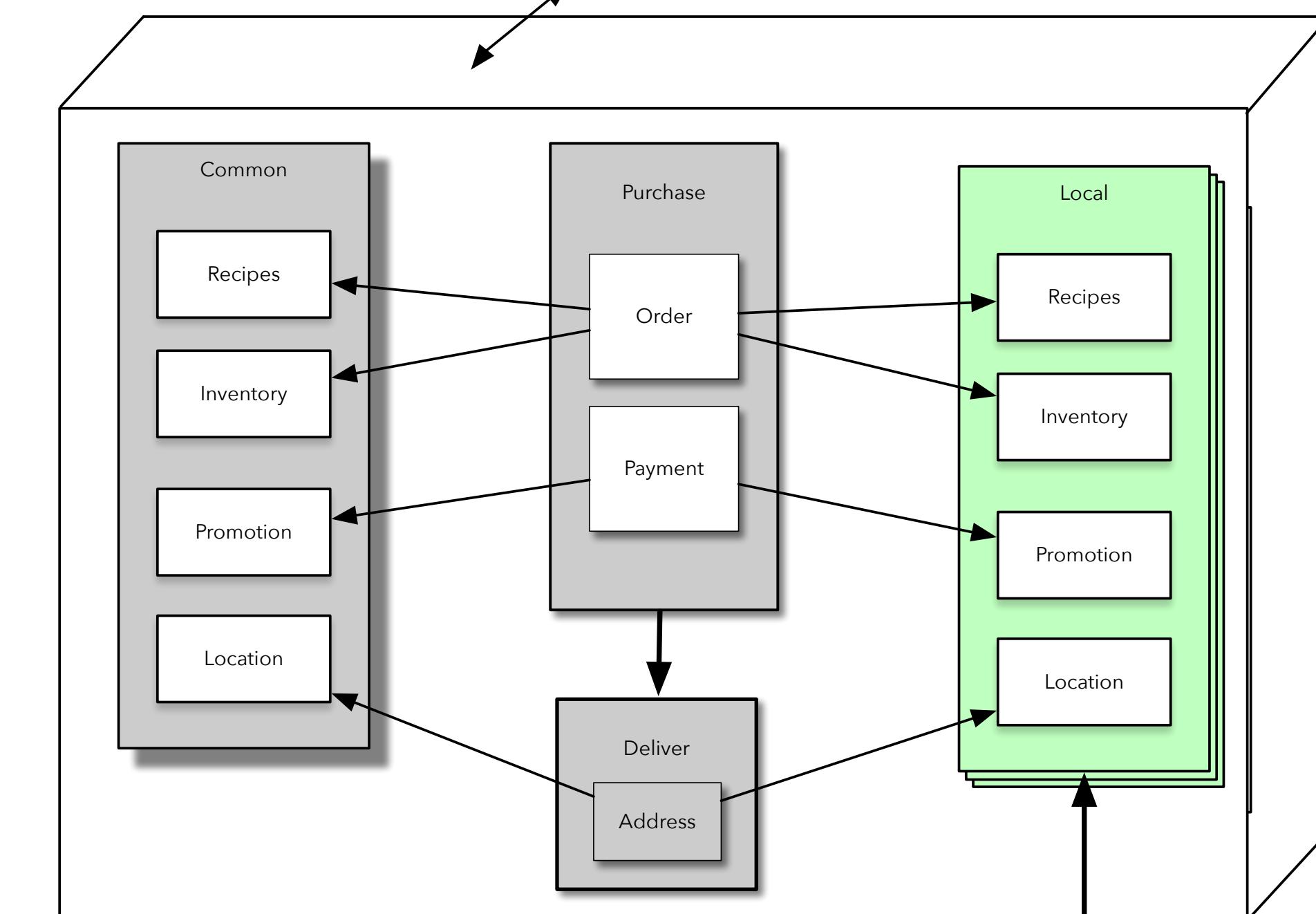
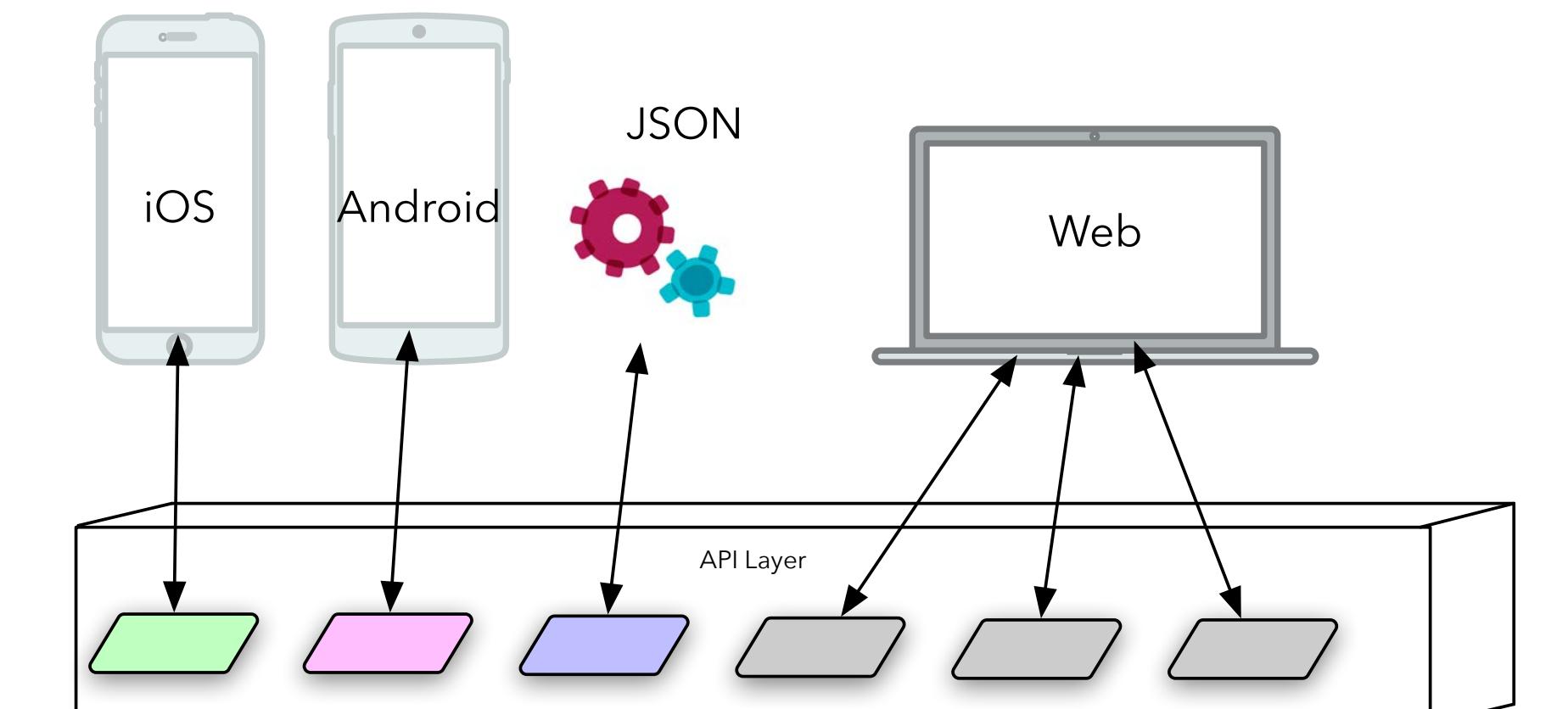
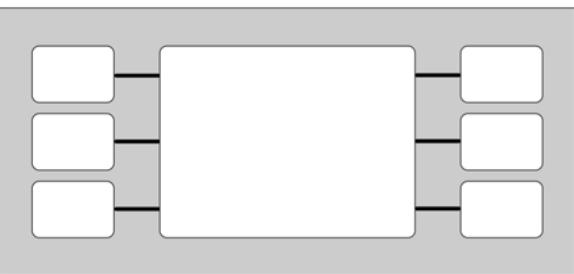
# Silicon Sandwiches



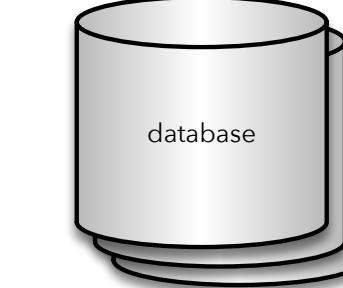
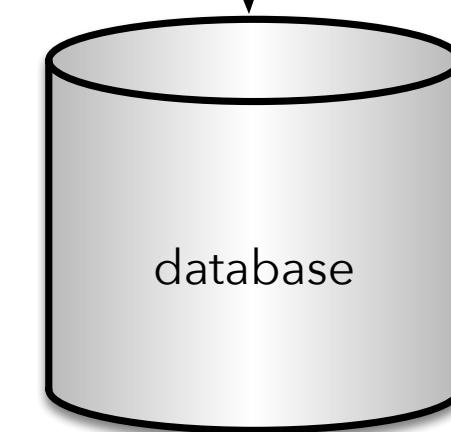
Customization handled by  
design, not architecture

Your Architectural Kata is...

# Silicon Sandwiches



Customization handled by architecture



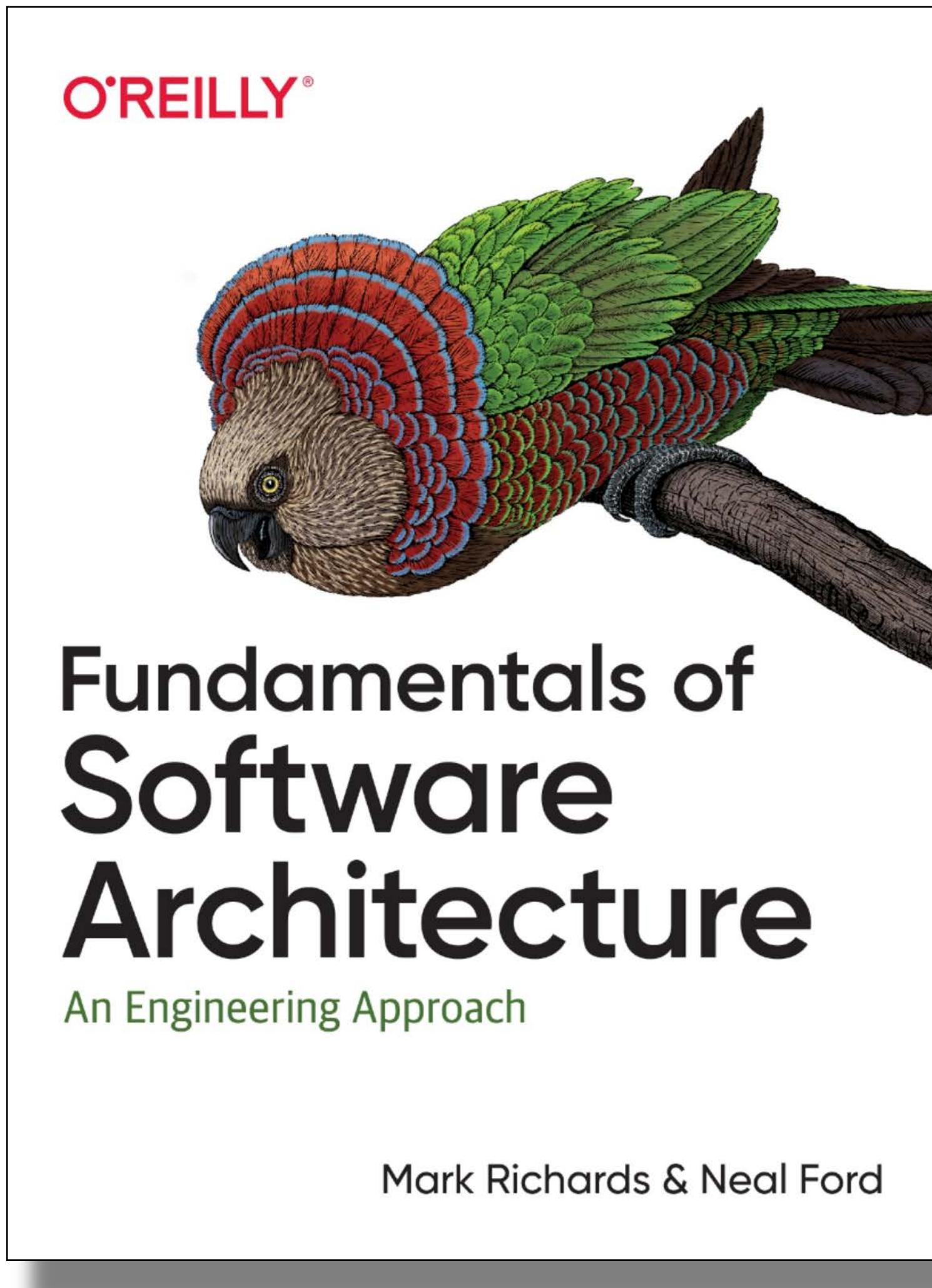
documenting  
software  
architecture



# documenting software architecture



# documenting software architecture



## Second Law of Software Architecture

*"Why is more important than how"*

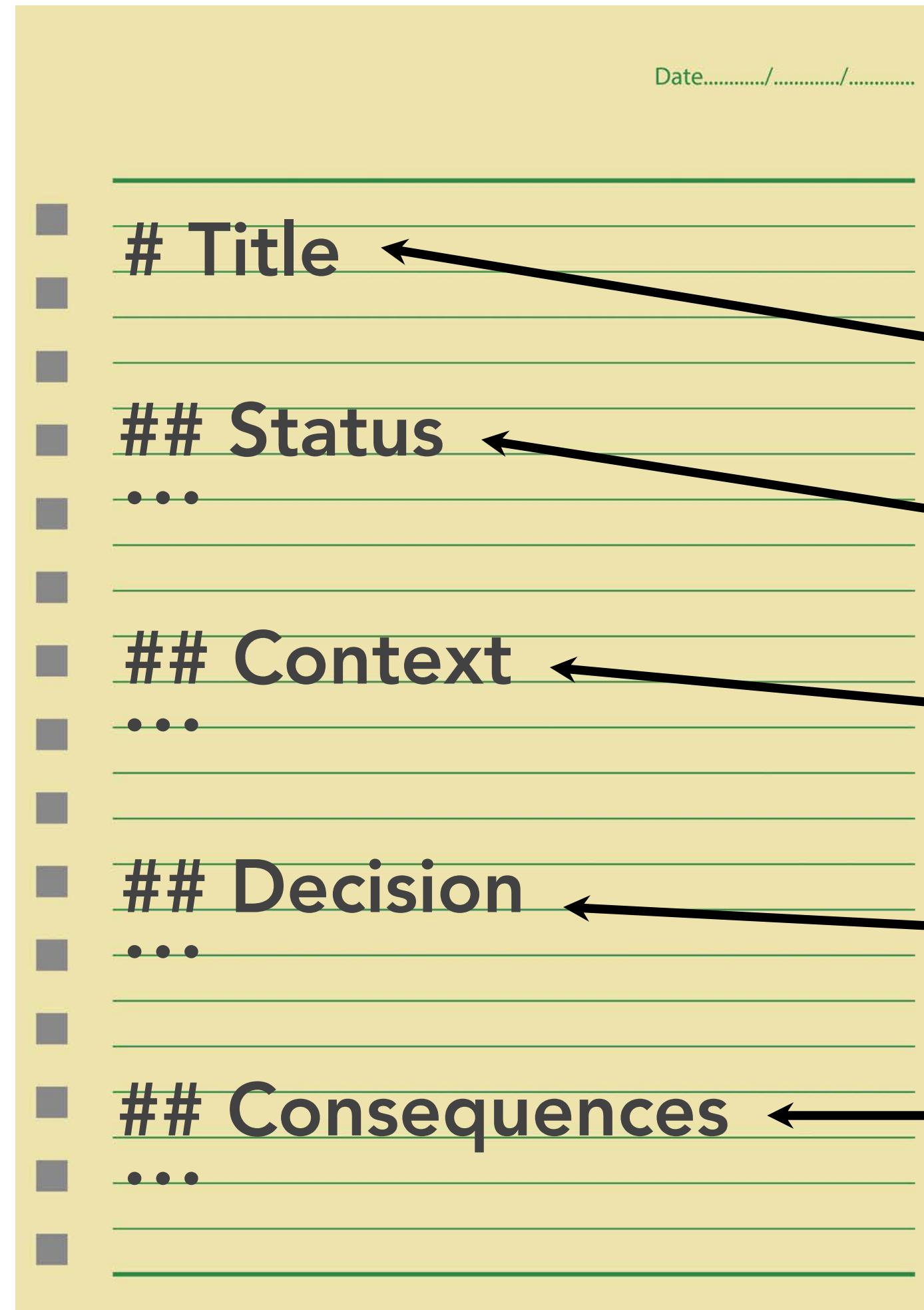
# documenting software architecture



“We will keep a collection of records for *architecturally significant decisions*: those that affect the structure, non-functional characteristics, dependencies, interfaces, or construction techniques.”

- Michael Nygard

# documenting software architecture



short text file; 1-2 pages long, one file per decision  
markdown, textile, asciidoc, plaintext, etc.

short noun phrase

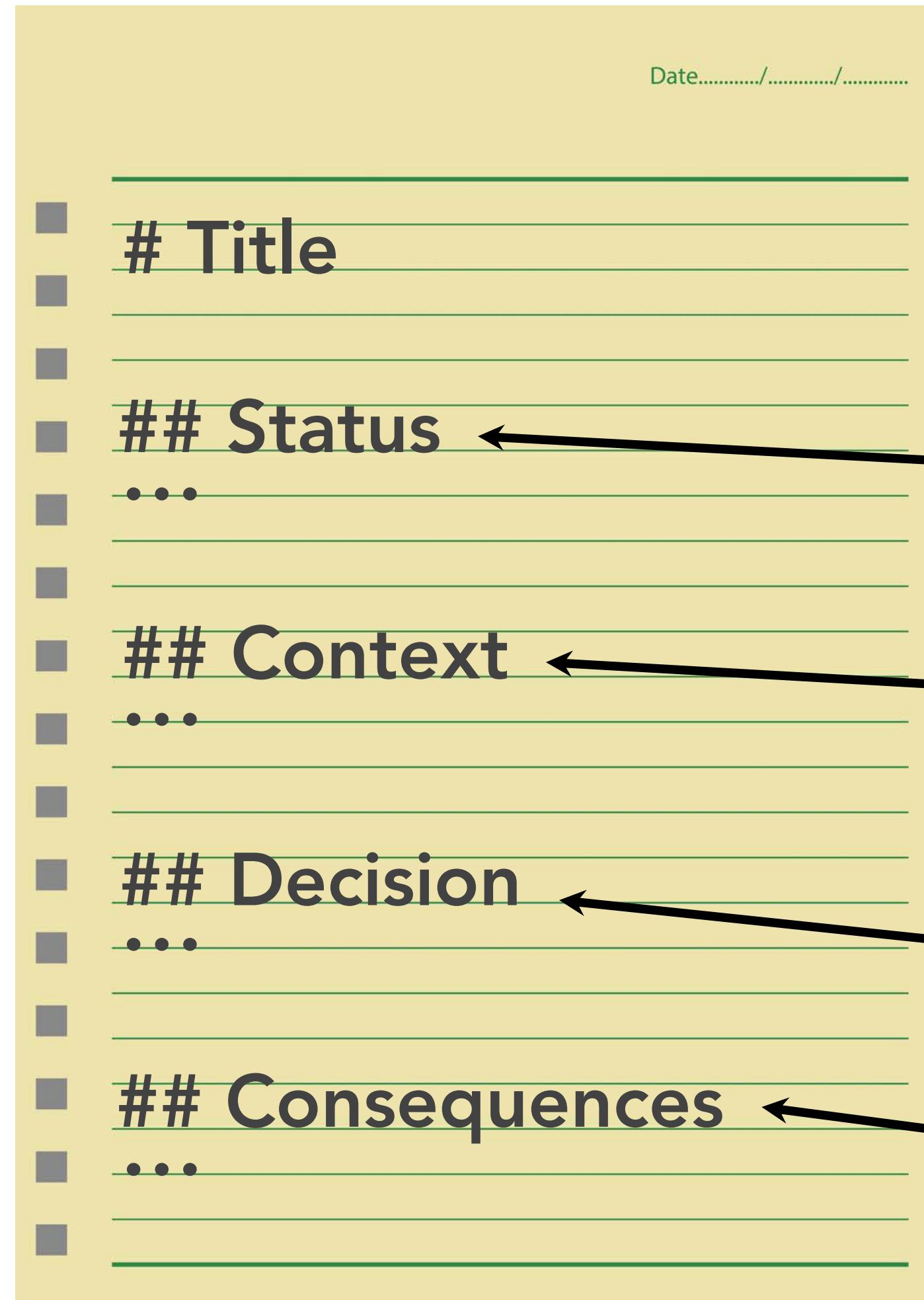
proposed, accepted, superseded

forces at play

response to forces

context after decision is applied

# documenting software architecture



short text file; 1-2 pages long, one file per decision  
markdown, textile, asciidoc, plaintext, etc.

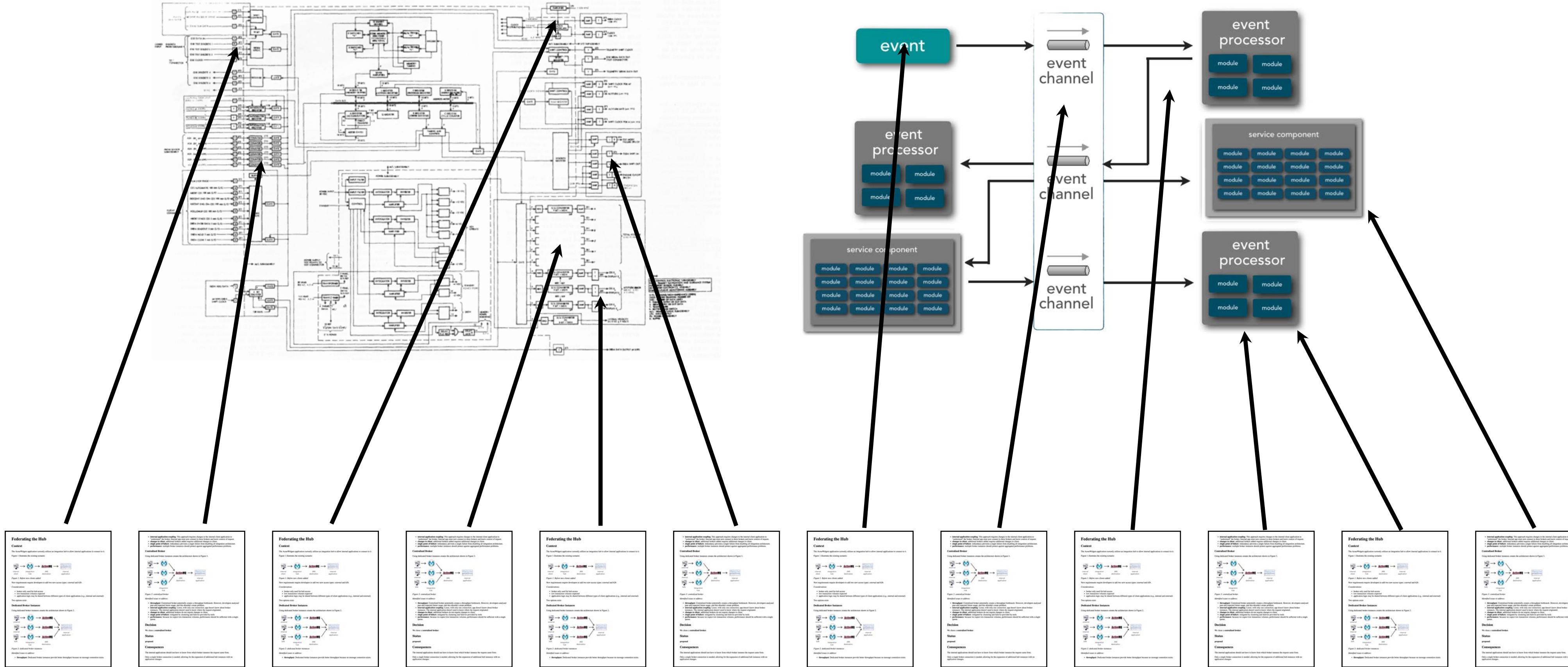
forces criteria for knowing when an architect must seek approval for a decision

description of the problem and alternative solutions available (documentation)

**justification (the “why”)**

**tradeoffs and impact of decision**

# documenting software architecture



Your Architectural Kata is...

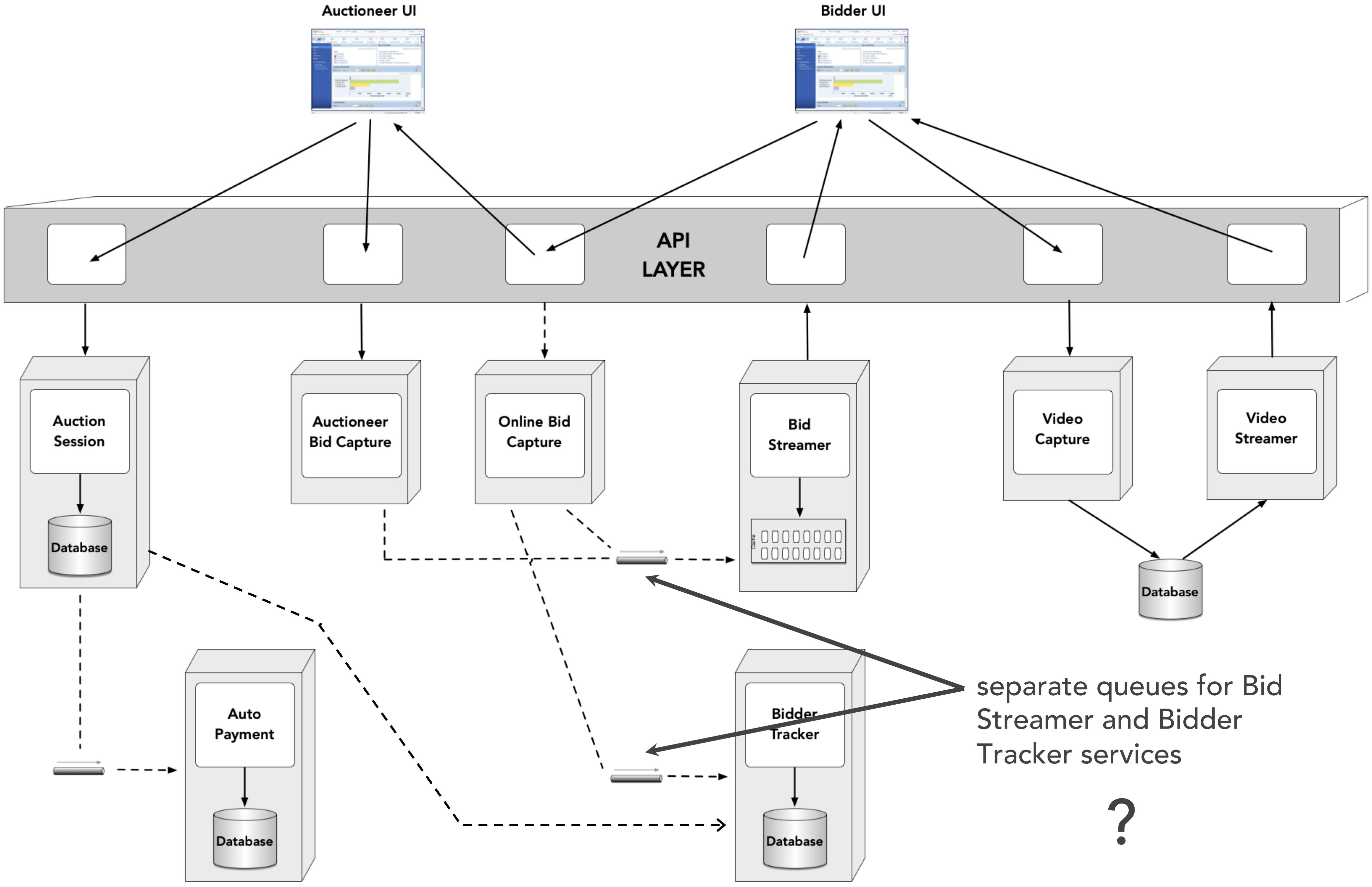
# Going Going Gone!

An auction company wants to take their auctions online to a nationwide scale--customers choose the auction to participate in, wait until the auction begins, then bid during the live auction as if they were there in the room, with the auctioneer.

- **Users:** scale up to hundreds of participants (per auction), potentially up to thousands of participants, and as many simultaneous auctions as possible
- **Requirements:**
  - bidders can see a live video stream of the auction and see all bids as they occur
  - auctions must be as real-time as possible
  - both online and live bids must be received in the order in which they are placed
  - bidders register with credit card; system automatically charges card if bidder wins
  - participants must be tracked via a reputation index
- **Additional Context:**
  - auction company is expanding aggressively by merging with smaller competitors
  - if nationwide auction is a success, replicate the model overseas
  - budget is not constrained--this is a strategic direction
  - company just exited a lawsuit where they settled a suit alleging fraud

Your Architectural Kata is...

# Going Going Gone!



# 1. Separate Queues for Bid Streamer and Tracker Services

## Status

---

Accepted

## Context

---

The Bid Capture Services, upon receiving a bid, must forward that bid to the Bid Streamer Service and the Bidder Tracker Service. This could be done using a single topic (pub/sub) or separate queues (p2p) for each service.

## Decision

---

We will use separate queues for the Bid Streamer and Bidder Tracker services.

Multiple bids will come in for the same ask amount. The Streamer service only needs the first bid received for that amount, whereas the Bidder Tracker needs all bids received. Using a topic (pub/sub) would require the Bid Streamer to contain logic to ignore bids that are the same as the prior amount, forcing the Bid Streamer to store shared state between instances.

The Bid Streamer Service stores the bids for an item in an in-memory cache, whereas the Bidder Tracker stored bids in a database. The Bidder Tracker will therefore be slower and might require back pressure. Using a dedicated Bidder Tracker queue provides this dedicated back pressure point.

## Consequences

---

This decision will require the Bid Capture services to send the same information to multiple queues.

# Use of Micro-kernel Architecture

## Status  
\_PROPOSED\_

## Context

Two key requirements of the system (\_promotions\_ and \_location services\_) have both global (affects all stores) and local (specific to location) requirements.

The current design features a modular monolith architecture, allowing individual stores to upload their behavior using JAR files, shown in *Figure 1*.

![modular monolith](fig1\_modular\_monolith.jpg) <br> \_Figure 1: the current state architecture\_

Currently, stores must specify custom behavior (product specials, promotions, location exemptions) via a JAR file, uploaded to the global site via FTP.

All local customizations reside in one service and in one set of tables in the master database.

To allow stores to most easily add and customize local behavior, the architects propose moving to a micro-kernel architecture, shown in *Figure 2*.

![microkernel architecture](fig2\_microkernel.jpg) <br> \_Figure 2: proposed microkernel architecture\_

The new design allows easy update of global policy (products, inventory, promotions) while allowing local stores to selectively those choices when appropriate.

## Decision

The architects decided to migrate the current monolithic architecture to a micro-kernel architecture.

## Consequences

The architects take advantage of the restructuring opportunity to localize databases to individual domains.

The new design also incorporates the BFF patterns, discussed in [004 BFF for device independence](#).

The new design will greatly improve the customization workflow.

- the local store plug-in architecture certifies customizations automatically
- promotions within threshold values go live within 15 minutes
- all stores work with generic workflows via the core system
- promotions
- location exemptions
- local products

## Use of Micro-kernel Architecture

### Status

**PROPOSED**

### Context

Two key requirements of the system (*promotions* and *location services*) have both global (affects all stores) and local (specific to location) requirements. The current design features a modular monolith architecture, allowing individual stores to upload their behavior using JAR files, shown in *Figure 1*.

*Figure 1: the current state architecture*

Currently, stores must specify custom behavior (product specials, promotions, location exemptions) via a JAR file, uploaded to the global site via FTP. Operations must certify the JAR, leading to delays in deploying new features.

All local customizations reside in one service and in one set of tables in the master database. Over time, as new customizations accrued, it has become a tangled mess.

To allow stores to most easily add and customize local behavior, the architects propose moving to a micro-kernel architecture, shown in *Figure 2*.

*Figure 2: proposed microkernel architecture*

The new design allows easy update of global policy (products, inventory, promotions) while allowing local stores to selectively those choices when appropriate.

### Decision

The architects decided to migrate the current monolithic architecture to become the core system for the new microkernel architecture, and build new functionality via plug-ins.

### Consequences

The architects take advantage of the restructuring opportunity to localize databases to individual domains. Communication between services now occurs via messaging.

The new design also incorporates the BFF patterns, discussed in [004 BFF for device independence](#).

The new design will greatly improve the customization workflow.

- the local store plug-in architecture certifies customizations automatically
- promotions within threshold values go live within 15 minutes
- all stores work with generic workflows via the core system, but locations can override to create custom behavior for:
  - promotions
  - location exemptions
  - local products

## Stat  
\_PROPO

## Con  
Two ke

## Status

The cur

### PROPOSED

![modu  
\_Figure

## Context

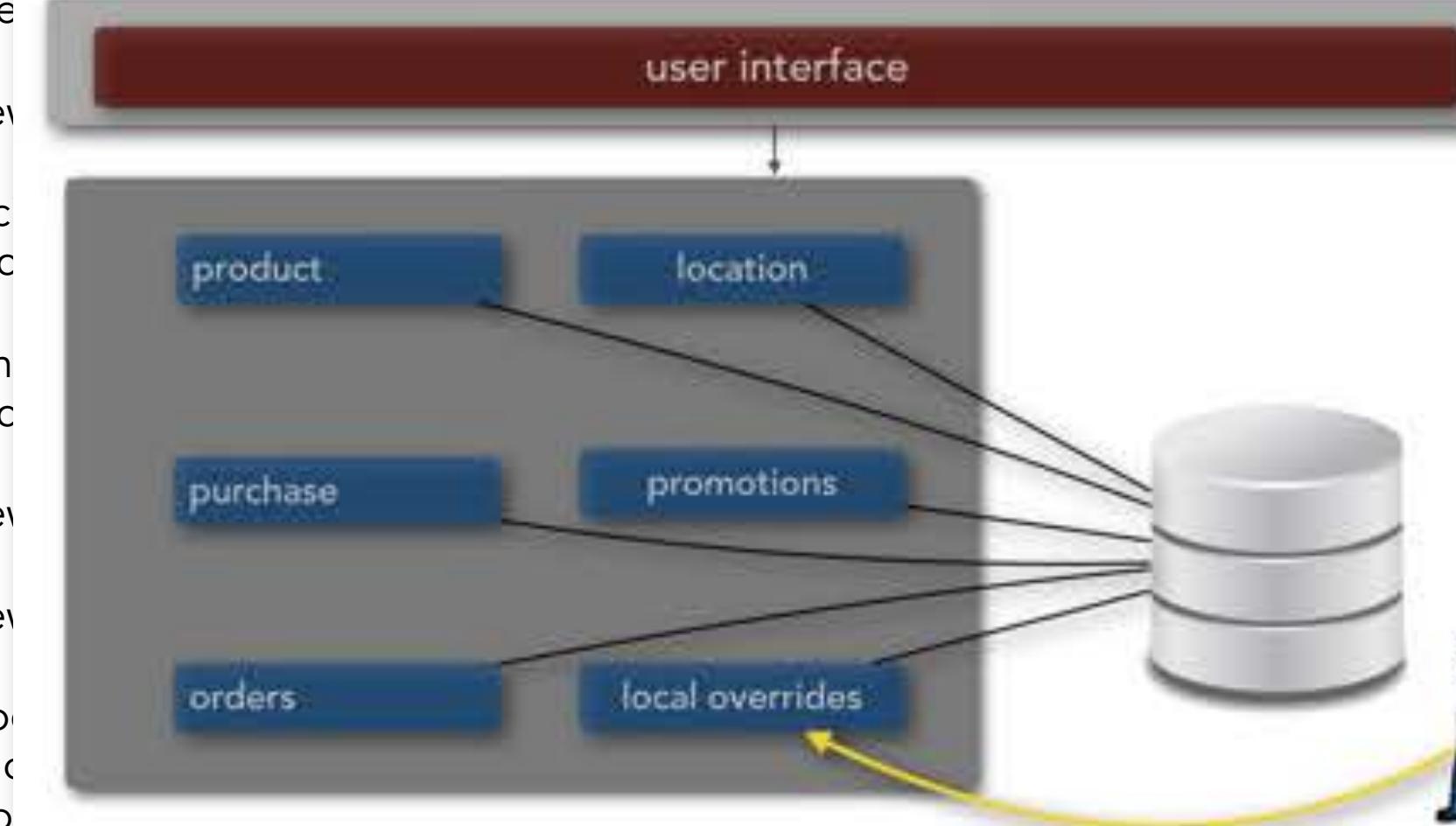
Current

All loca Two key requirements of the system (*promotions* and *location services*) have both global (affects all stores) and local (specific to location) requirements.

To allo

The current design features a modular monolith architecture, allowing individual stores to upload their behavior using JAR files, shown in *Figure 1*.

![micro  
\_Figure



*Figure 1: the current state architecture*

Currently, stores must specify custom behavior (product specials, promotions, location exemptions) via a JAR file, uploaded to the global site via FTP. Operations n the JAR, leading to delays in deploying new features.

All local customizations reside in one service and in one set of tables in the master database. Over time, as new customizations accrued, it has become a tangled me

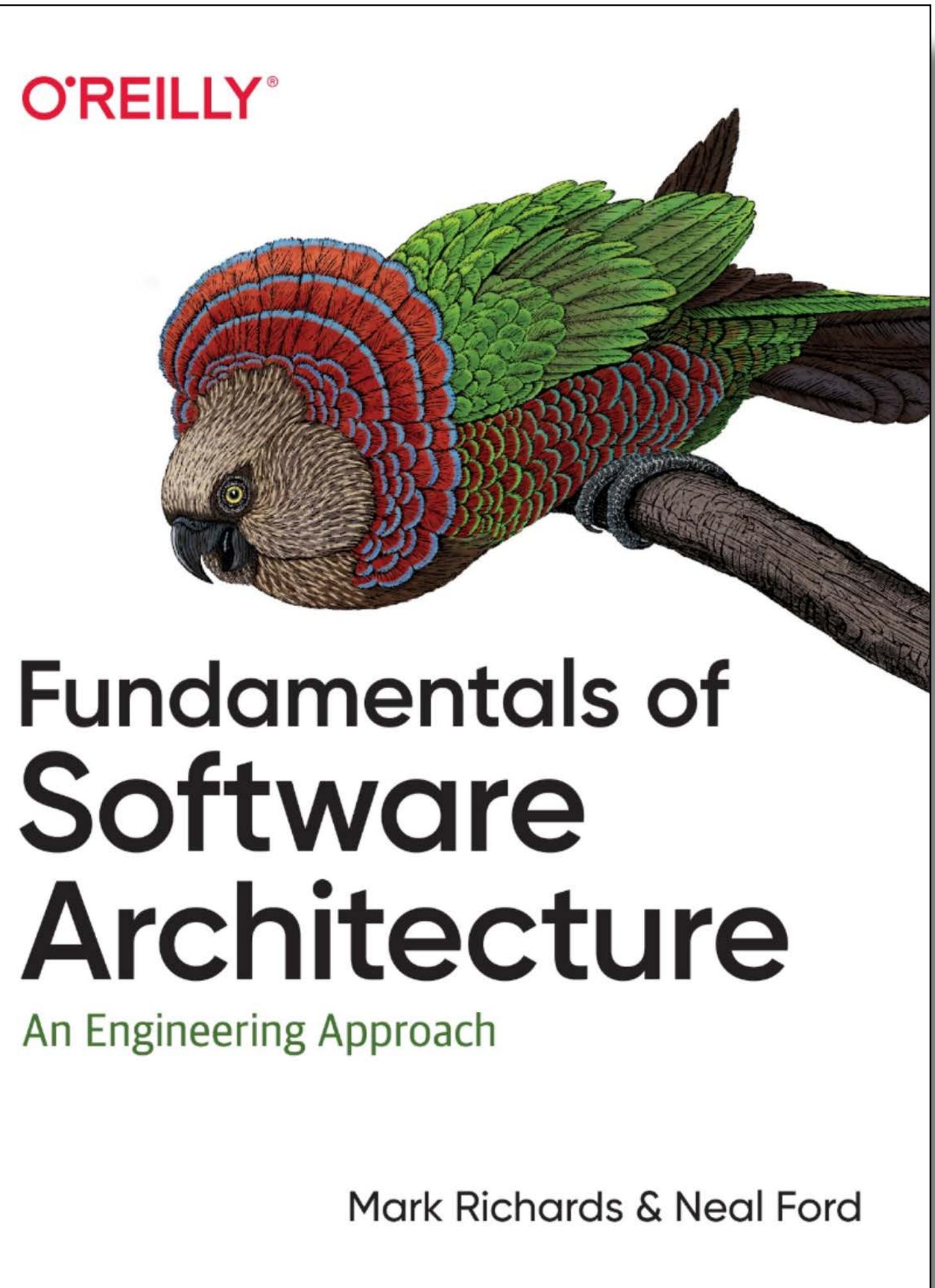
# summary



# summary

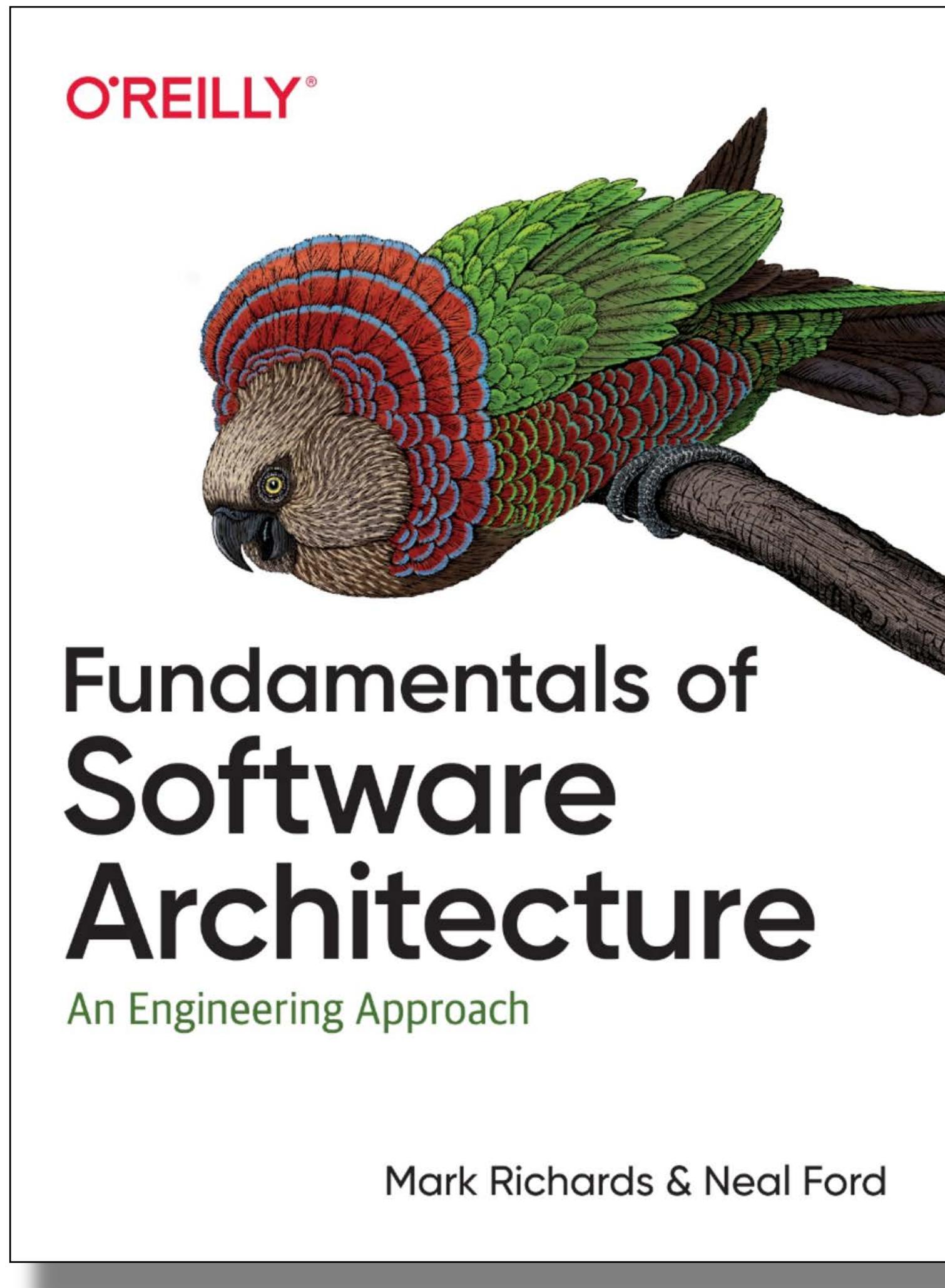
## First Law of Software Architecture

*“Everything in software architecture is a tradeoff”*



There are no right or wrong answers in architecture; rather, it's always about **tradeoffs**

# summary



## Second Law of Software Architecture

*"Why is more important than how"*

every architecture decision should be accompanied with a **technical and business justification**

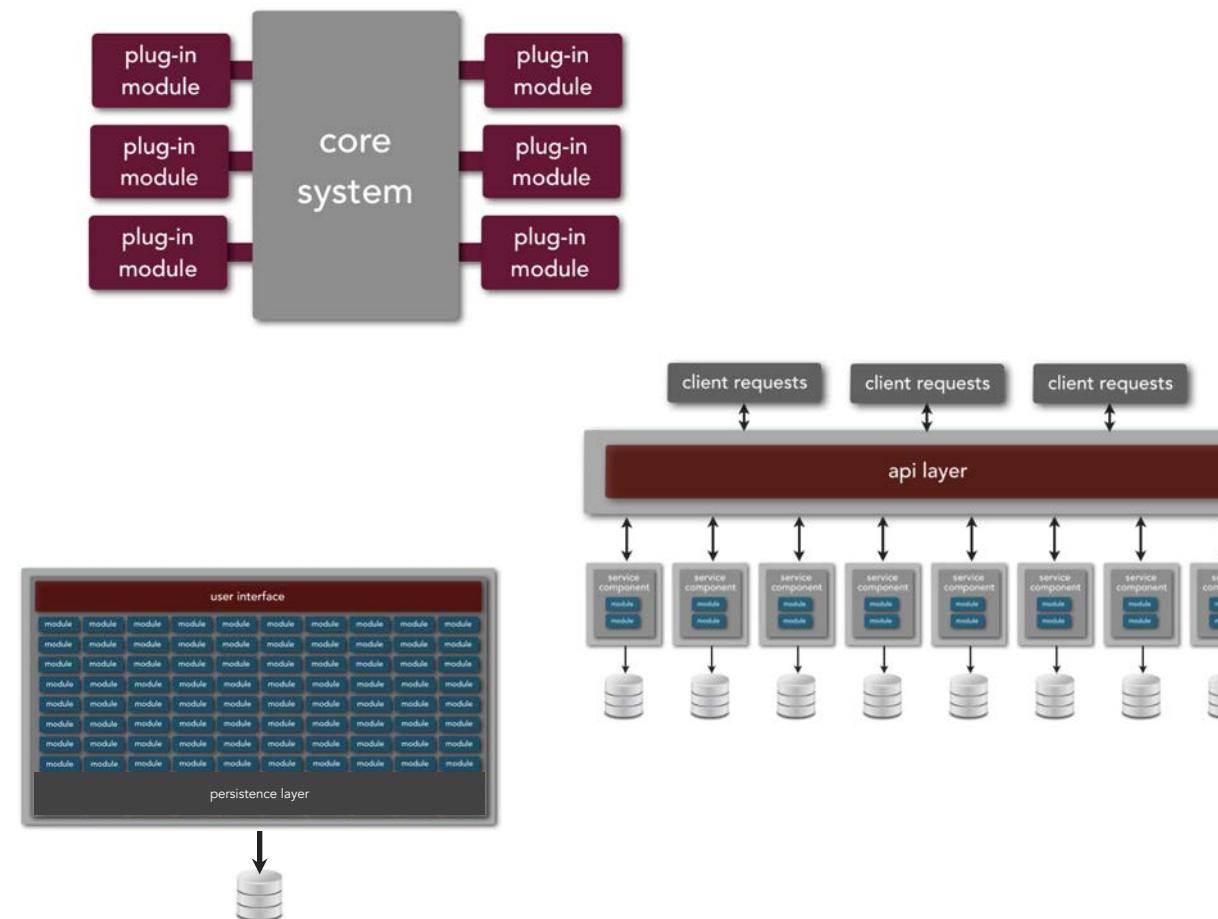
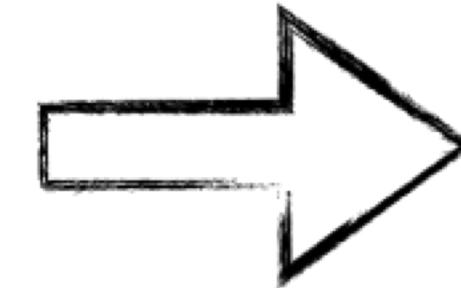
# summary



scalability

reliability

availability



***business needs and architectural characteristics*** are what drive the architecture!

# more resources



Not Secure — [fundamentalsofsoftwarearchitecture.com](http://fundamentalsofsoftwarearchitecture.com)

[fundamentalsofsoftwarearchitecture.com](#)

[Architectural Katas](#) [Architectural Katas](#) [Fundamentals of Software Architecture](#) [List of](#)  
[Architecture Katas](#)

---

## Architectural Katas

*inspired by Ted Neward's original [Architectural Katas](#)*

"How do we get great designers?  
Great designers design,  
of course."  
Fred Brooks

"So how are we supposed to get great architects, if  
they only get the chance to architect fewer than  
a half-dozen times in their career?"  
Ted Neward

---

## About

Architectural Katas are intended as a small-group (3-5 people) exercise, usually as part of a larger group (4-10 groups are ideal), each of whom is doing a different kata. A Moderator keeps track of time, assigns Katas (or allows this website to choose one randomly), and acts as the facilitator for the exercise.

Each group is given a project (in many ways, an RFP—Request For Proposal) that needs development. The project team meets for a while, discovers requirements that aren't in the original proposal by asking questions of the "customer" (the Moderator), discusses technology options that could work, and sketches out a rough vision of what the solution could look like. Then, after they've discussed for a while, the project team must present their solution to the other project teams in the room, and answer challenges (in the form of hard-but-fair questions) from the other project teams. Once that challenge phase is done, the room votes on their results, and the next project team takes the floor.

## Rules

Doing an Architectural Kata requires you to obey a few rules in order to get the maximum out of the activity. [Read Rules »](#)

## Rules

The rules are broken down by the different Phases of the exercise. However, one rule trumps all the others: **Any other questions that are not already covered by these rules, you may ask the Moderator about.** When in doubt, ask.

**Preparation: Getting your project team together**

The first step is to assemble your project team. There are only a few rules regarding the composition

# more resources



The screenshot shows the ThoughtWorks website homepage with a pink header. The header includes links for ESPAÑOL, PORTUGUÊS, DEUTSCH, 中文, a search bar, and navigation buttons for NEWS, EVENTS, and CONTACT US. Below the header, the ThoughtWorks logo is displayed, followed by a menu with links to Clients, Services, Products, Insights, About us, and Careers. A large banner features a red and purple diagonal striped background with a white soundwave icon and the word "PODCASTS". Below the banner, the text "Tech Talks that matter" is displayed, followed by a descriptive paragraph about the podcast's focus on tech topics and trends.

ESPAÑOL PORTUGUÊS DEUTSCH 中文

Search thoughtworks.com

NEWS EVENTS CONTACT US

ThoughtWorks®

Clients Services Products Insights About us Careers

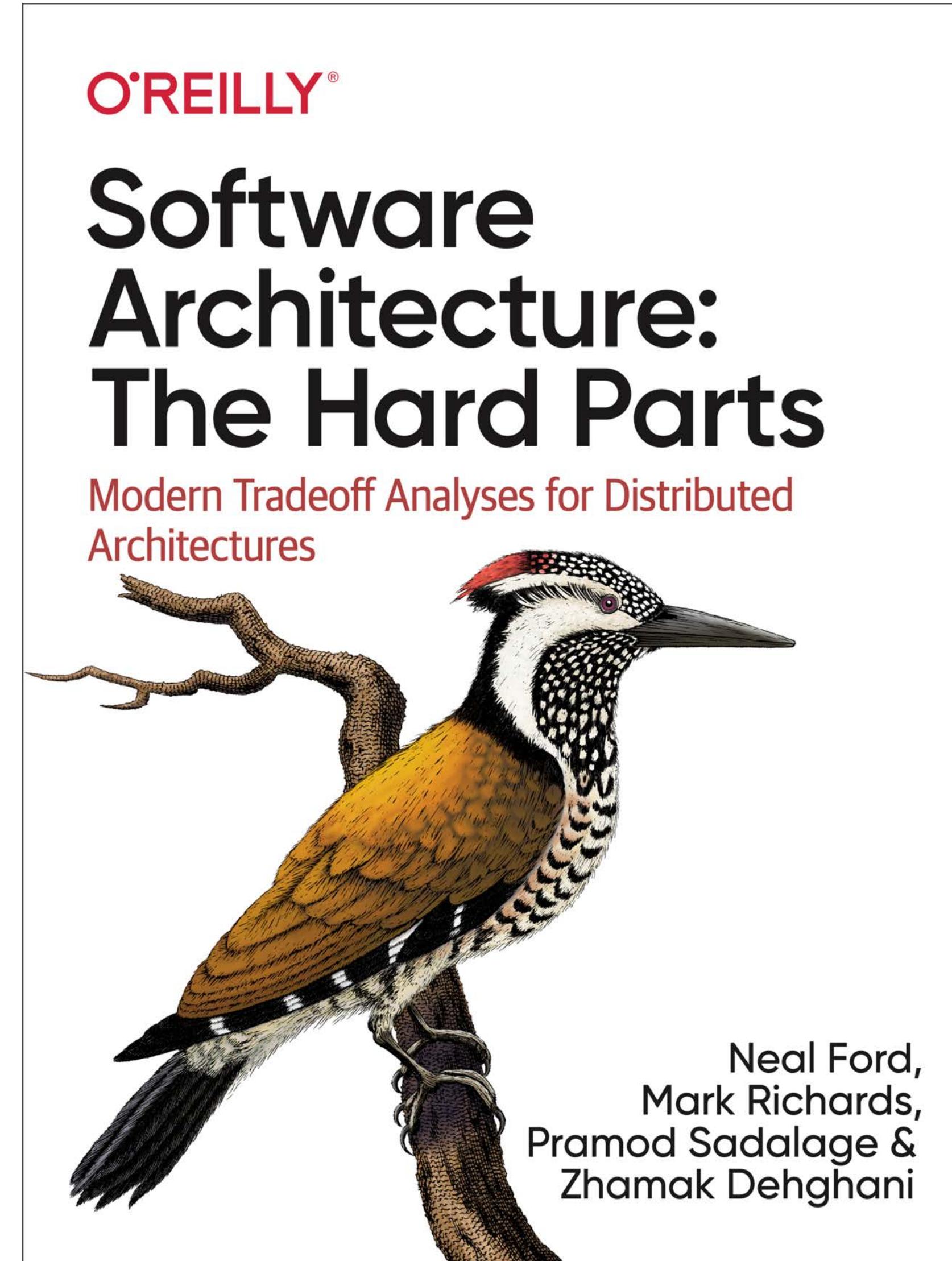
PODCASTS

Tech Talks that matter

The ThoughtWorks podcast plunges deep into the latest tech topics that have captured our imagination. Join our panel of senior technologists to explore the most important trends in tech today, get frontline insights into our work developing cutting-edge tech and hear more about how today's tech megatrends will impact you.

<https://www.thoughtworks.com/podcasts>

Coming  
soon...



<http://architecturethehardparts.com/>

# [DeveloperToArchitect.com](https://www.developertoarchitect.com)

HOME TRAINING LESSONS ARTICLES BOOKS VIDEOS FORUM UPCOMING EVENTS ABOUT ME

## Developer to Architect

Training and resources for the journey from software developer to software architect  
Mark Richards, Software Architect and Founder

### Software Developer To Software Architect

*"The journey from developer to software architect is a difficult and uncharted path filled with lots of challenges, pitfalls, and confusion. The purpose and goal of DeveloperToArchitect.com is to provide resources and training to help you along the journey to becoming an effective software architect"*

[Mark Richards](#), Software Architect, Founder of DeveloperToArchitect.com

I created this website to provide developers with resources and guidance in the long and difficult journey from software developer to software architect. In here you'll find helpful lessons, articles, books, videos, source code, and training classes I teach.

[Software Architecture Monday](#) is a free bi-weekly video lesson series on some aspect of software architecture. These 10 minute YouTube videos contain various aspects of application, integration, and enterprise architecture.

#### Contact Me

To contact me regarding any public and private software architecture training classes I offer, you can reach me (Mark Richards) directly at [info@developertoarchitect.com](mailto:info@developertoarchitect.com)

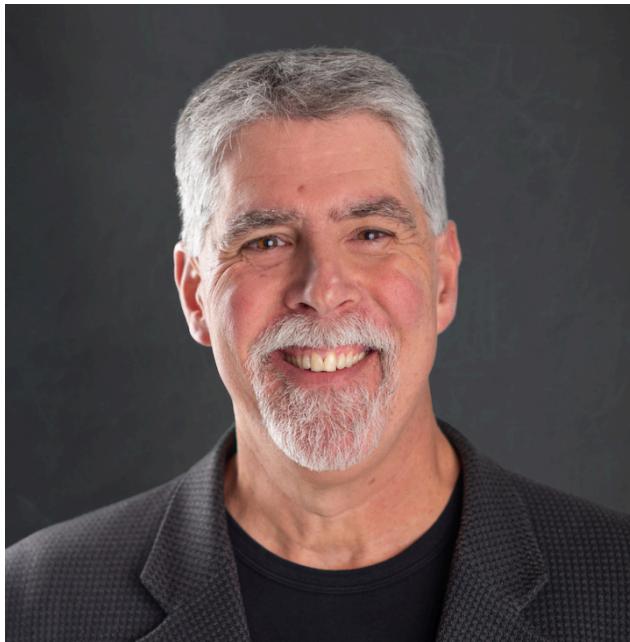
#### Public Live Virtual Training

I'm excited to announce that during our current social situation I am conducting live, hands-on virtual workshops. See my [Training](#) page and [Upcoming Events](#) page for a complete listing of classes, details, dates, and how to register.



go do some  
architecture!

# Architecture by Example



## Mark Richards

Independent Consultant

Hands-on Software Architect, Published Author

Founder, [DeveloperToArchitect.com](http://DeveloperToArchitect.com)

<http://www.wmrichards.com>

@markrichardssa



## Neal Ford

ThoughtWorks

Director / Software Architect / Meme Wrangler

<http://www.nealford.com>

@neal4d



O'REILLY®